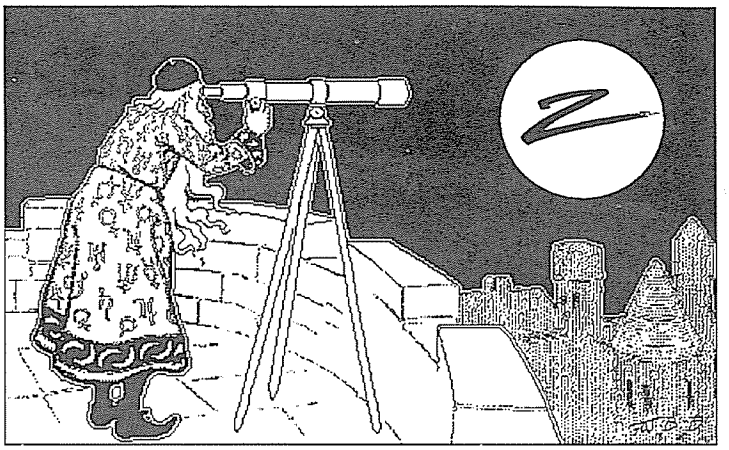
Z™

THE NEWSLETTER FOR ZBASIC PROGRAMMERS

SPRING-SUMMER 1987, Volume 1, Issue 2



INSIDE...

Michael A. Gariepy
Editor in Chief
Advertising

Andrew Gariepy
Technical Editor

CORPORATE
Zedcor, Incorporated
PRESIDENT
Andrew Gariepy
CHAIRMAN
Michael A. Gariepy

Spring-Summer 1987

"Z" is published quarterly by Zedcor, Inc., 4500 E. Speedway, Suite 22, Tucson, AZ 85712-5305. (602) 881-8101. Support line: (602) 795-3996. Subscription rates are $19.95 a year, $37.00 for two years (add $5 for overseas Canada).

To subscribe by phone call toll free: **1-800-482-4567**.

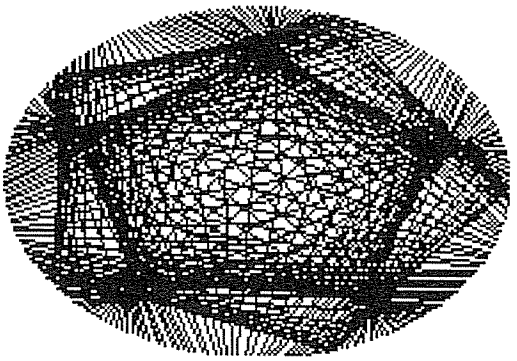
Please send address changes to the Zedcor address above.

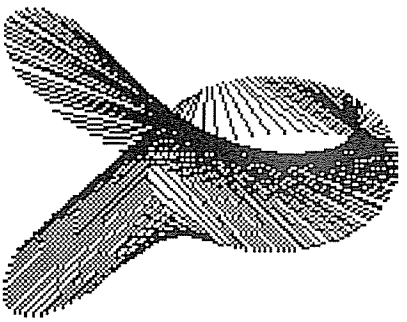
Advertising offices are at 4500 E. Speedway, Suite 22, Tucson, AZ 85712-5305. 1-800-482-4567

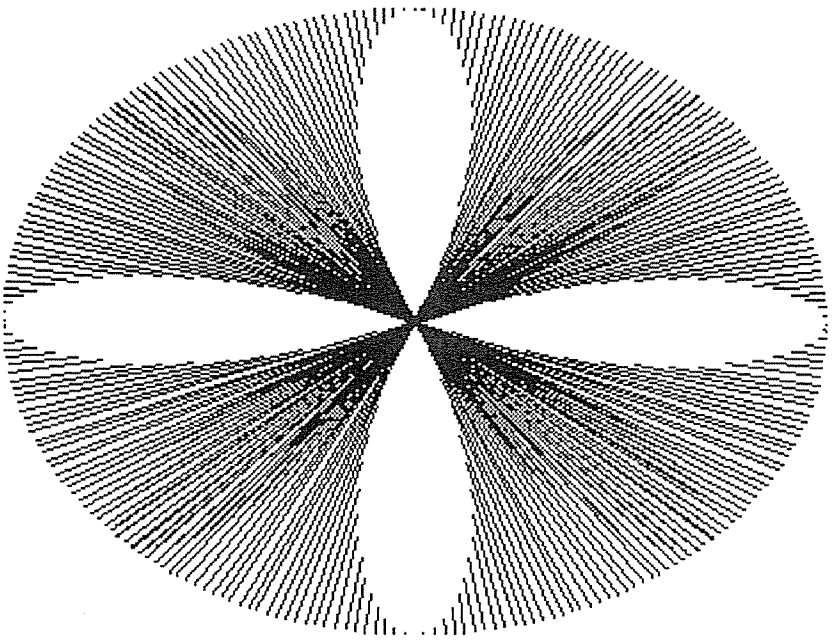
"Z" is a technical journal not affiliated in any way with computer manufacturers.

CREATE INCREDIBLE RANDOM SHAPES

DEMONSTRATES THE SPEED AND POWER OF ZBASIC'S INTEGER SINE AND COSINE (USR8 and USR9)

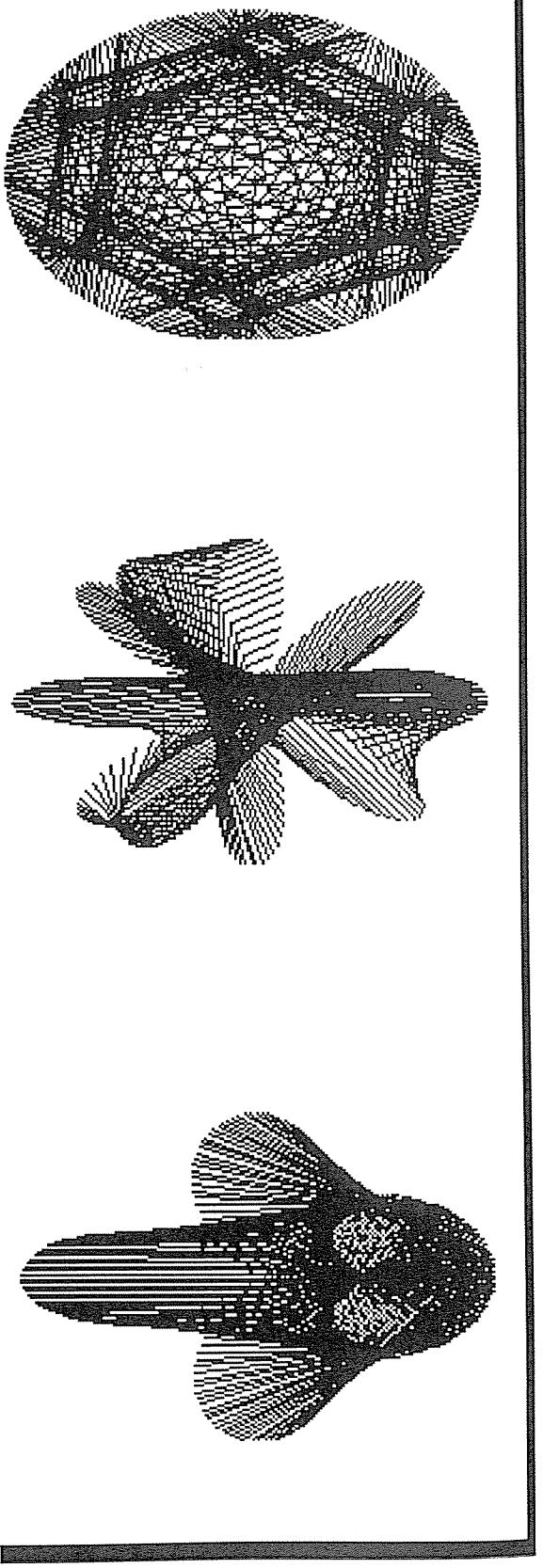






Next Page...

```
CLS:RANDOMIZE
:
REM  SHOWTYME
REM  by James Ducasse
REM  (Modified for speed by
REM  Andrew Gariepy)
:
REM  This is an excellent example
REM  of using the high-speed
REM  USR8 and USR9 INTEGER
REM  SINE and COSINE functions
REM  built into all versions of
REM  the ZBASIC compiler.
:
CR%=7: V1=4: TITLE%=1
:
DO
  FOR R=0 TO 256
    TRONX
    A%=USR9(Q*R)/4
    X1=(USR9(R)*A%)/50+512
    Y1=(USR8(R)*A%)/50+384
    B=R+E
    B2%=USR9(V1*B)/4
    X2=(USR9(B)*B2%)/50+512
    Y2=(USR8(B)*B2%)/50+384
    PLOT X1, Y1 TO X2,Y2
    I$=INKEY$:IF LEN(I$) THEN "END"
  NEXT R
  Q=RND(12): IF Q=7 THEN Q=0
  IF Q=1 AND V1=0 THEN V1=RND(12)
  E=RND(200)
  V1=RND(12): IF V1=7 THEN V1=0
  IF V1>6 THEN V1=V1-12
  IF Q<1 AND V1=1 THEN V1= RND(50)
  IF Q=1 AND V1=1 THEN V1=RND(100)
  CLS: CR%=RND(17)
  IF CR%=2 THEN CR%=4
  IF CR%=3 THEN CR%=6
  IF CR%>6 AND CR%<10 THEN CR%=CR%+4
  CR1%=CR%+1:DELAY 2000
UNTIL LOOP
:
"END": CLS
END
```



"LIFE"

FOR ALL VERSIONS OF ZBASIC

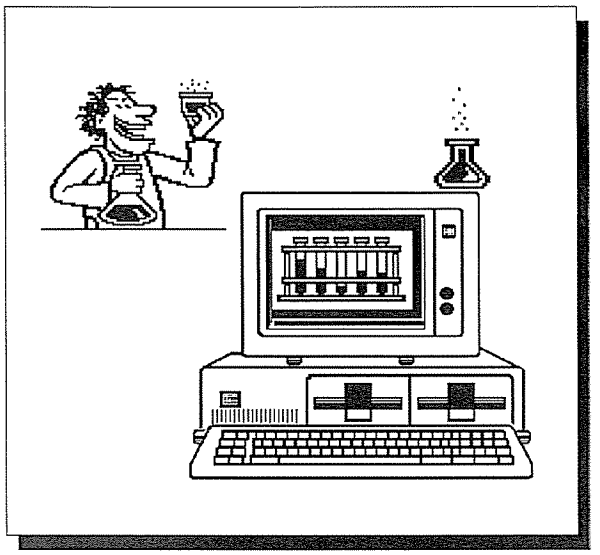
This program creates life forms based on the starting pattern that you give it (and it's great fun).

The original idea was created by a guy named "Conway" that appeared in an issue of "Scientific American" some years back. Have fun modifying the code as you see fit. A few graphic additions could make this program really great.

```
00010 CLS : PRINT TAB(18);"LIFE" : GOSUB 640
00020 H=70:V=22:X1=1:Y1=1:X2=V:Y2=H
00030 DIM A(22,70), 80 B$(22),UBOUND(10):C=1:G=1
00040 LINE INPUT "-> ";B$(C)
00050 IF B$(C)="DONE" THEN B$(C)="" : GOTO 70
00060 C=C+1 : GOTO 40
00070 C=C-1 : L=0 : CLS
00080 FOR X=1 TO C-1
00090   IF LEN (B$(X))>L THEN L=LEN(B$(X))
00100 NEXT X
00110 REM FIND EDGES OF COLONY
00120 X1=INT((V-X)/2)
00130 Y1=INT((H-L)/2)
00140 REM COUNT POPULATION
00150 FOR X=1 TO C
00160   FOR Y=1 TO LEN(B$(X))
00170     IF MID$(B$(X),Y,1)<>" " THEN A(X1+X,Y1+Y)=1:P=P+1
00180   NEXT Y
00190 NEXT X
00200 REM  PRINT OUT SCREEN
00210 LOCATE 0,0 : CLS LINE
00220 PRINT "GENERATION:";G;" POPULATION:";P;
00221 IF I9 THEN PRINT " INVALID";
00230 X3=V:Y3=H:X4=1:Y4=1:P=0
00240 G=G+1
00250 FOR X=1 TO X1-1: PRINT
00251 NEXT X : REM BLANK LINES ABOVE COLONY
00260 TRONX
00270 FOR X=X1 TO X2 : REM VERTICAL AREA OF THE COLONY
00280   PRINT : CLS LINE
```

continued next page...

```
00290   FOR Y=Y1 TO Y2 : REM HORIZONTAL AREA OF THE COLONY
00300     IF A(X,Y)=2 THEN A(X,Y)=0 : GOTO 380
00310     IF A(X,Y)=3 THEN A(X,Y)=1 : GOTO 330
00320     IF A(X,Y)<>1 THEN 380
00330     PRINT TAB(Y);"*"; : REM PRINT ONE CELL
00340     IF X<X3 THEN X3=X
00350     IF X>X4 THEN X4=X
00360     IF Y<Y3 THEN Y3=Y
00370     IF Y>Y4 THEN Y4=Y
00380   NEXT Y
00390 NEXT X
00400 REM EVOLVE & CHECK
00401 REM FOR GROWTH PAST EDGES
00410 X1=X3:X2=X4:Y1=Y3:Y2=Y4
00420 IF X1<3 THEN X1=3:I9=-1
00430 IF X2>(V-2) THEN X2=(V-2):I9=-1
00440 IF Y1<3 THEN Y1=3:I9=-1
00450 IF Y2>(H-2) THEN Y2=(H-2):I9=-1
00460 P=0
00470 FOR X=X1-1 TO X2+1
00480   FOR Y=Y1-1 TO Y2+1
00490     C=0
00500     FOR I=X-1 TO X+1
00510       FOR J=Y-1 TO Y+1
00520         IF A(I,J)=1 OR A(I,J)=2 THEN C=C+1
00530       NEXT J
00540     NEXT I
00550     IF A(X,Y)=0 THEN 590
00560     IF C<3 OR C>4 THEN A(X,Y)=2:GOTO 580
00570     P=P+1
00580     GOTO 600
00590     IF C=3 THEN A(X,Y)=3:P=P+1
00600   NEXT Y
00610 NEXT X
00620 X1=X1-1:Y1=Y1-1:X2=X2+1:Y2=Y2+1
00630 GOTO 210:REM PRINT NEXT GENERATION
00640 PRINT
00641 PRINT "   ENTER A STARTING DESIGN OF ASTERISKS."
00650 PRINT "   TYPE A PERIOD FOR A LEADING SPACE."
00660 PRINT "   USE A MAXIMUM OF ONE LINE,
00661 PRINT "   PRESS 'ENTER' TO END THE LINE.":GOTO 680
00670 PRINT "   PRESS 'RETURN' TO END THE LINE."
00680 PRINT "   BE SURE CAPS LOCK IS ON,"
00681 PRINT "   AND ENTER 'DONE' WHEN YOU ARE READY.": PRINT
00690 RETURN
00700 END
```



USER SUBMISSIONS DISAPPOINTING

We are currently compiling a list of many routines and utilities that you have sent us and that we have been putting together here at Zedcor.

While there have been a few submissions the overall response has been less than we hoped. Please feel free to send us listings or diskettes with your subroutines so we can make them available to everyone.

We are currently putting together listings for Cross-Reference utilities, math and scientific functions. Others will follow. As we complete and de-bug these routines we will make them available in this newsletter and eventually in direct mailings.

Many thanks to the folks that have made submissions. Your support is deeply appreciated by us and your fellow ZBasic programmers.•••

NEW LOGO

Zedcor has introduced it's new ZBasic logo. The new Z was designed by a local Tucson artist, Karen Scott.

We have been looking for a good logo for a number of years. Our first attempts were crude and only made Zenith mad. We are delighted to finally have a logo that fits the product.

Many thanks to Karen for her brilliant design.•••

Z BASIC™

NEW EDITORIAL LAYOUT

The layout of this newsletter has been altered from the first edition. Admittedly, the premier edition was our first attempt at creating a newsletter.

There will be no more confusing breaks in the text flow and the sections will be kept together for better readability.

Remember, this is only our second newsletter and I'm sure there's room for improvement. Please send us your comments.

Many thanks to all those readers that suggested changes to the first newsletter. •••

SUPPORT BLUES

Hi, this is Jeff in tech support! *How may I help you today?*

Jeff Moore (or Less)

Oh Wow man! This is really great! You guys really know how to support your customers man! Hey, I'm having a real problem integrating my old Atari Cobol programs into my ZBasic code. **What gives?** Like is the interface old fashioned or what? I called Bored Land and they...

... and my dog died right there in front of me. I'll tell ya, I nearly cried for a month. So I looked in the users manual for the IBM 370x and guess what? Nothin there at all ... not even a hint.

Oh yeah, does ZBasic let you put Pascal commands directly into your program like the Wang Eunich PCX does? **Why not?!?!** If only you could do that you'd have the best BASIC version of Pascal ever. Did you see the Game on TV last night? **Boy howdy! was it exciting!!** I swear the team is getting better every year!

Did you guys ever write a program to simulate Soccer? I did back in 1953 on an old FDS 123 bowling machine...

What's with This Guy? *Doesn't he know that other people are waiting for support too?* **Boss says I have to be nice...**

Yes sir. Sir I must get off now Sir!

...and low and behold the computer burned like a bonfire in the night. I tried to douse the flames with brillo pads but the sucker just burned even brighter!

Did I ever tell you about my third wife? She was a really sweet-heart. I would have kept her cept she was more interested in playin' round with the neibor's gardner. I never could understand what makes a woman tick. They just don't make sense to me.

Were you ever involved with women? Of course not! You're far to good at computing and much to logical to let some woman get her hands on your wallet! By the way...

did you ever learn how to use the LONG IF function in ZBasic. I went to a users group in Spokane back in 85, they gave a good description of how that stuff works.

I swear I'll get around to using that command one of these days! Hows the weather in Tucson these days? I was there back in 41 during the way... boy ... Did you guys ever write a program to simulate Soccer? I did back in 1953 on an old FDS 123 bowling machine...Did you guys ever write a program to simulate Soccer? I did back in 1953 on an old FDS 123 bowling machine...Did you guys ever write a program to simulate Soccer? I did back in 1953 on an old FDS 123 bowling machine... gram to simulate Soccer? I did back in 1953 on an old FDS 123 bowling machine...Did you guys ever write a program to ffff simulate Soccer? I did back in 1953 on an old FDS ffgg g ggggg sssyy ybdfg dd rr 123 bowling machine...gram to simulate Soccer? I did back in 1953 on an old FDS sdf 123 bowling machine...Did you guys ever write a program to simulate Soccer? I did not did back in 1953 on an old FDS 123 bowling machine...gram to simulate Soccer? I did dfgg back in 1953 on an old FDS 123 bowling machine...Did you guys ever write a program to fddf

I'm Going To HANG UP NOW SIR!!

I can't take it anymore!!

Please shut-Up! Is this a recording? Do you hear me? Good Bye! ... **Bye!**

<click>

Far away in another State...

Bzzst... Bsszzt... BBrsszzt.. Bsszzt.. Bzzsstt.. Bssztt

Gosh Darn Al-mighty! Zedcor's support line is busy again.

I haven't had too much trouble getting through before. I'll try again in a few minutes.

Half-hour, 4 phone calls and a Migraine Headache later...

BUSY AGAIN!!! OK you Slime Buckets! I've got a serious problem! If you don't answer soon *I may switch to Micro Sloth Products!!!*

Bzzst... Bsszzt... BBrsszzt.. Bsszzt.. Bzzsstt.. Bssztt

I've got a mind to Call the BBB, FTC, FBI, YWCA, KGB *and the NAACP!*

AT LAST, I FINALLY GOT THROUGH! *I'll teach these people!!*

I'M MAD AS HELL!

That's right! You're support line is always busy and I'm sick of it!

It's a good thing I like you guys...

now here's the problem I'm having with my program....

***What did I do to* DESERVE THIS?**

VERSION 4.0 RELEASED FOR MSDOS, MACINTOSH AND APPLE //

Zedcor has released all new versions of ZBasic for MSDOS, Macintosh and Apple // computers. Registered owners were advised of the upgrades in April and May.

If you did not receive these notices and you are a registered owner of ZBasic, please contact Zedcor offices to confirm that we have you on our mailing list. We work hard to keep our mailing list updated but occassionally change of addresses and such are not recorded.

The MSDOS version has many new features like Hercules and EGA graphics, large variable size, SELECT CASE structure, Full Screen Editor and lots more. See the MSDOS section of this newsletter for details.

Version 4.0 for the Macintosh has many new additions and enhancements.

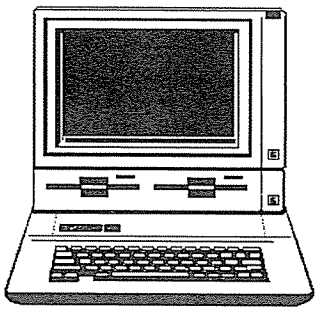
It now comes with SELECT CASE, a completely new editor window, 128K ROM toolbox support (including TE and LIST functions) a newly revised and comprehensible manual (without addendums to the addendums) and much more. See the Macintosh section of this newsletter for details.

Apple // ProDOS version was released in early June and has so many more features than the old DOS 3.3 version that we won't even attempt to describe them here.•••

USER GROUP NEWS

An overwhelming number of "Z" subscribers are interested in a ZBasic user's group. We are putting this information together now and will attempt to include a questionaire in the next issue.

The questionaire will consist of ideas we have (and some subscribers have suggested) to make the users group a success. Nevertheless, it is YOUR Users group so we'll attempt to make it to your specifications. •••



Apple // ProDOS™ and DOS 3.3 versions

The ProDOS version of ZBasic is finally out. Shipments began about June 1st (if you don't have your upgrade yet you better call Zedcor).

The Product is better than we ever anticipated. Functionality far surpasses any other BASIC for Apple // systems.

The next issue of "Z" will contain special notes and information about the new ZBasic 4.0 ProDOS version.

For those people interested in getting support for the ProDOS version you will be happy to know that Jeff Moore, Zedcor's support person, owns an Apple //c and can help you with any questions you may have. He was our main "Beta-Tester" for this version too.

Greg Branche, product manager for the Apple // versions, has also been appointed the product manager for the MSDOS/IBM versions of ZBasic. He is an important part of our team and we give him our congratulations.

Problem with DOS 3.3 MOUSE on //c.

A "new" Apple //c was announced by Apple Computer on September 15, 1986. This //c+ added memory expansion capabilities to the former //c configuration. The ROM subroutines for the memory expansion are mapped to slot four, the former location of the mouse interface. The mouse has been moved to slot seven. Unfortunately, the DOS 3.3 version of ZBasic expects the interface to be in slot four. Consequently, when a MOUSE(0) function call is made, the system dies a horrible death.

I tried to patch the runtime package to find the mouse interface, no matter which slot it is in, but there just isn't enough free space to fix it. However, you can patch it from within your ZBasic program, IF the program is running on a //c+.

If your program is using the mouse, and you are writing the program to work on any Apple, you should include the following lines as part of your intialization sequence:

```
LONG IF  PEEK(&FBB3)=6
     AND PEEK(&FBC0)=0
     AND PEEK(&FBBF)=3
  POKE &D1F8, &7F
  POKE &D1FF, &7F
  POKE &D204, &7F
  POKE &D20A, &FF
  POKE &D20F, &FF
  POKE &D217, &C7
  POKE &D21C, &C7
  POKE &D21E, &70
  POKE &D222, &C7
END IF
```

This first checks to see if it's running on an Apple //c+, and if it is, then it patches the runtime module to use the interface in slot seven. If it's not in a //c+, the runtime module will remain as is, and will use the interface in slot four (the 'standard' location).

The ProDOS version of ZBasic will correctly scan the slots to identify the location of the mouse interface, no matter which slot it's in. In addition, the DOS 3.3 version will be corrected in the next release.

NOTE TO APPLE //GS USERS

On page D8 of the ZBasic ProDOS appendix, item 2B explains how to create a disk that will boot directly into ZBasic. The manual says to copy the "ProDOS" file to a freshly formatted disk from a ProDOS system disk (such as the disk that was included with your //GS). The actual file is really "P8" (P16 is ProDOS 16). Copy this file to your ZBasic master diskette and rename it "ProDOS".

LETTER FROM GREG BRANCHE TO PRODOS USERS OF ZBASIC 4.0

First of all, I'd like to thank you for returning the comment sheets. Without input from you, I would never have found some of the bugs that were hiding in the system. I also appreciate the other comments and suggestions that were made, some of which have been implemented in this final version. (At least, final for now. No software is ever complete.)

I have attached a list of the bug fixes and other additions that have been made to the system. If there is something that is not on the list which you have experienced with the pre-release software, please try the final version of the software before you start calling me nasty names. Chances are, the problem has already been found and fixed, but I just never made a note of the problem for inclusion in the list.

And now, for my reply to some common comments that were made:

First of all, it seems like everybody asked for more memory. My reply: C'mon gang! Y'all knew that the 128K version was forthcoming! I knew that there wasn't a lot of room in the 64K version; that's why I've been working my tail off on the 128K version. Now that you've got the 128K version in your hot little hands, maybe I'll hear less of this complaint. Speaking of memory, a lot of people asked for added commands to the system, such as a COPY command, accessing drives by drivespec (as opposed to pathnames), etc. Remember; everything that I add to ZBasic uses memory. If MY software uses up the memory, that means that YOUR software won't have access to it. I tried to make the best compromise possible between memory use and system power.

...a lot of people asked for added commands...

Remember; everything that I add to ZBasic uses memory. If MY software uses up the memory, that means that your software won't have access to it.

One of the compromises that I made was in the graphics departments. I figured that once the 128K version was released, about the only people who would be using the 64K version were those who only had 64K (does that sound logical?). Since the machine would only have 64K, then double-hi-res graphics would not be supported by the hardware. Since double-hi-res wasn't being supported by the hardware, then there was no reason for the software to support it. So, in the 64K version, MODE 7 is now equal to MODE 5.

Another compromise was in the use of drive specifications (such as S6, D1) as opposed to pathname specifications (/ZBASIC.128K). If I had allowed drive specifications to be used in commands (or, more so, within programs), it would have increased the complexity and memory usage of the system. ProDOS will ONLY access disks by use of pathnames. BASIC.SYSTEM allows drivespecs because it converts the drivespec into a pathname under the

surface. If you want ZBasic to do the same thing for you, I have to ask first: How much memory are you willing to sacrifice for this convenience? (One other point to keep in mind: All of the newer software that I've seen doesn't allow drivespecs either.)

Many people asked for a multiple-line Cut & Paste in the full screen editor. I'm currently working on it. Look for it in a future release. (That's called planned obsolesence. You're all familiar with that, aren't you?)

Many people asked for a screen dump facility within ZBasic. I would really like to help, but I can't. There is a great variety of printers and interface cards out there in Apple land, and there is just no way that ZBasic could support all of the various methods that are required to translate a graphic picture into a series of dots on paper (unless, of course, you want me to use up some more precious memory for the system...).

What I would suggest is that you get an interface card that has screen dump software built into it's firmware. A couple of good examples of this are the Grappler card, and the Fingerprint card.

A few people asked for ProDOS VAR files, which are accessed by the BASIC.SYSTEM commands STORE and RESTORE. Basically (pun intended), you have the same thing in ZBasic. It's just a little hidden. First, make sure that the variables that you want to STORE are all grouped together in memory. (DIMension all them together at the beginning of your program.) Then, find the beginning address of the block (using the VARPTR function). Next, find the ending address of the block of variables, then use the BSAVE function to save the block of variables out to memory. When you wish to RESTORE the variables, just locate the beginning address of the block of variables again, then use the BLOAD function to read them back in.

continued next page

ELIZABITCH* and FRIENDS

We were kindly sent a copy of this software for Apple // DOS 3.3 systems (written in ZBasic) and have to admit it was very interesting.

It won't win awards for manners, since it was designed with Rudeness and Abuse in mind.

For you Apple // folks that enjoy being insulted and are into other such nasty things, this program may be for you (of course we would never endorse such a thing).

To order; send $7.00 to Karl Bunker, 321 S. Huntington Ave, Boston, MA 02130.

Karl also has another program for those people that "Hate" computer adventure games. It's called "Terminal Boredom". He says "it is a thoroughly silly, illustrated, adventure game spoof". Available for $7.00 at the address above.

(This is all that BASIC.SYSTEM does, I think, but it just does most of the work for you).

Many people asked for more examples, math functions available on disk, etc.

Well, I have included more examples on the disk. In addition, Jeff (our tech support guy here at Zedcor) has put together a disk-full of AP-PENDable functions that do everything but wash your car for you. Call our order line for price and availability.

There are those of you that would like to see the full screen editor commands match those of Appleworks. I'd like to accomodate you, but with the frequency of the "look and feel" law-suits coming out of Cupertino these days, I'd hate to step on somebody's toes. (You get my meaning, I'm sure.)

There were a few comments like "I wish that the 80-column editor had the Wordstar commands like the 40-column editor has" and "I wish I had the choice of either the 40 or 80 column editor, since I have a small monitor". Good news, gang. You can fool the system. If you prefer the 40-column editor over the 80-column version, simply put the 40-column editor on your boot disk, and rename it to "FSEDIT.80.OBJ". This should please everybody (including ZBasic). This won't work the other way, though. In order to use the 80-column editor, you MUST have a //e with the aux-slot 80-column card, a //c, or //GS. It won't work on a][+ or //e with an expansion slot 80-column card (such as Viewmaster).
Sorry, but the aftermarket cards won't let me get to their video memory for scrolling purposes.

Now I'd like to clear up some misunderstandings that have appeared in the comment sheets. First of all, a couple of people commented on the SIEVE program not running because there wasn't enough memory. If you CONFIGure the number of file buffers to 0, then there will be enough memory to run the program. Remember, each file buffer that is reserved uses up 1K of memory. That memory can be used ONLY as a file buffer. If your program doesn't use files, then don't reserve any file buffers.

The MODE command does not intentionally clear the screen for you. On the 64K version, when going into an 80-column mode, the screen IS cleared, but this is a function of the 80-column firmware and not ZBasic. With the 128K version, I was able to get around this limitation (since I knew EXACTLY which 80-column card is in the machine). The reason I didn't have the system clear the screen for you (like Applesoft's HGR) was so that you could flip back and forth between graphics and text without losing anything on either screen. (No more POKE-16304,0.) If you want to clear the screen when switching modes, simply issue a CLS command immediately after the MODE x.

INSLOT and OUTSLOT were never intended to access interface cards that attempt to use the Applesoft ROM directly. Clock cards are good examples of this kind of interface card. Take this Applesoft sequence, for example:

```
PRINT D$;"PR#4"
PRINT D$;"IN#4"
INPUT TM$
REM  GETS TIME STRING FROM
REM  THE CLOCK CARD
PRINT D$;"PR#0"
PRINT D$;"IN#0"
PRINT TM$
```

Lines 10 and 20 redirect output and input to/ from the card. Line 30 tells the card to fill TM$ with an ASCII string representing the current time. Actually, it should return the string to Applesoft 1 character at a time, but what it actually does is go out and find the address of TM$ directly, then fill it with the ASCII string. It terminates by returning an ASCII carriage return code to Applesoft, which then sees this as

the end of the input. This obviously won't work under ZBasic, since ZBasic strings are stored in an entirely different manner than Applesoft strings. The only alternative is to read the clock at a machine language level. A function is included on the disk which does this for the Clockworks clock card (might work for the Timemaster card too).

Many people have gotten into the habit of loading their programs into a /RAM disk prior to running it (for an increase in speed), and ZBasic will allow this. Just remember that ZBasic will attempt to find a /RAM disk to load itself into during boot-up, and if that /RAM disk is the same one that it is currently residing in, it will happily copy itself over itself. If you still want to load ZBasic into a /RAM disk manually, make sure that there is no other /RAM disk online prior to running ZBasic.

Those of you with a GS have had a problem getting color in MODE 7. I've had the same problem, but it's not in ZBasic. When the desktop software is run on the GS, it flips a softswitch that turns off the color-burst signal for double-hi-res (so that you can read what's on the desktop). When you run another application from the desktop, the desktop software doesn't reset the color-burst signal, so it appears that color is not available. The solution: enter the control panel, and make sure the display is set to color. Actually, if you already have it set to color, just entering the control panel will flip the softswitch.

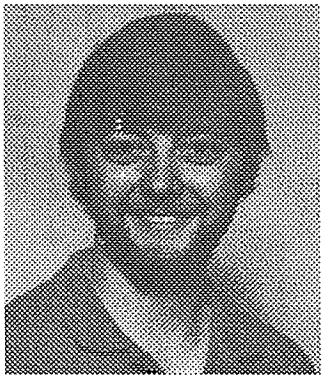
Mention was made of the runtime libraries. Let me explain the need for them now. ZBasic is not a linking compiler. When you RUN* a two line-program, such as:

```
CLS
PRINT "Hello"
```

you get the entire runtime library (I'm looking into building a linking version of the system). I split most of the runtime library out into separate files so that if you had more than one stand-alone program, you wouldn't be wasting disk space with multiple copies of the libraries. If I were to combine everything into one file (as was done with the DOS 3.3 version of ZBasic), the stand-alone programs would be unnecessarily large. This is another one of those compromises that I made.

And now, a bit of bad news. The IIGS graphics disk that was mentioned in the ProDOS appendix is vaporware. We will not be offering it as a product. Sorry.

That's about all I can think of for now. Again, thanks for giving us your support. I hope that this final version can live up to your expectations (although I don't really think it will. Apple users usually set their goals so high...). As I stated in the letter that you received with version 3.95, I want to make this the BEST BASIC available on the Apple //'s. I think that if ZBasic isn't there yet, it sure is close. Thanks for your help in the fine-tuning phase of the project. Let me know if there's anything else that I can do for you.



Greg Branche
Product Manager
Apple/IBM Software Division

FIXES AND ADDITIONS

Following is a list of features added and fixes peformed to ZBasic version 3.95
to make it version 4.0, release date 6/1/87.

ADDITIONS

✳ The LOCATE statement now allows either, or both, coordinate expressions to be optional. Run the RUN.ME program for full details.

✳ More documentation has been included on converting Applesoft programs to ZBasic.

✳ More example programs have been included on disk.

✳ Instructions have been included on converting DOS 3.3 ZBasic programs over to the ProDOS disk format.

✳ PREFIX can now be used in place of PATH as an editor command.

✳ The order of the arguments in the LOCATE command can now be configured by the user.

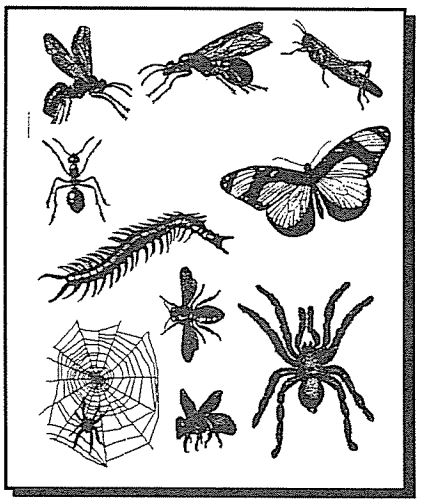
✳ The full screen editor can now be entered by the use of single keystroke commands. Run the RUN.ME file for more information.

✳ Instructions have been included in the RUN.ME program for setting up a large aux-slot ram card as a /RAM disk.

✳ The full-screen editor Solid-Apple-0 (NEW) command now requires verification before proceeding.

EITHER AN ADDITION OR A BUG FIX. I DON'T KNOW WHICH!

• If ZBASIC.HLP can't be found in the system directory, the editor will now look for it in the currently logged directory. All you have to do is use the PATH command to set the prefix to the directory that contains the help file.



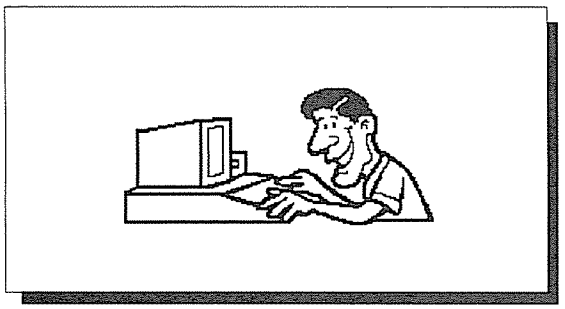
BUG FIXES

◆ The editor now ensures that a file specified in the "RUN pathname" command is a ZBS file, and that it exists, prior to passing control to the compiler.

◆ A configured "CONVERT TO UPPERCASE" option will now be saved correctly.

◆ LIST "label" will not hang if "label" does not exist.

◆ The line editor will no longer accept the right arrow in the EDIT command. This was causing problems with 80-column cards when the right arrow was echoed to the screen, switching the screen from 80 to 40 columns.



◆ PATH command will no longer allow removal of the default prefix.

◆ Autotab now works correctly in the Full Screen Editor (FSE).

◆ Using the TAB key to move the cursor to the right margin in the FSE no longer causes the system to crash.

◆ Deleting a line when the last line in the file was on the screen no longer causes the last line to disappear.

◆ Solid-Apple-5 (Find Next) now saves any changes to the line prior to proceeding.

◆ The FSE no longer loses characters on the screen when being used on an Apple //c.

◆ Open-Apple-4 (Insert Line) crash fixed.

◆ Long programs are no longer truncated when transferring from the line editor to the FSE. A warning is issued, and the command is aborted.

◆ The Lazer 128 should now recognize the screen funtion codes sent out by the FSE.

◆ CHAINing now works as specified in the manual.

◆ DEFxxx now recognizes lower-case variables as variable specifiers.

◆ MODE 7 now displays in color with Apple (Video 7) compatible RGB cards. (I hope. I don't have an RGB card to test with. Could somebody verify this for me please?)

◆ A stand-alone program now boots up as expected and does not give the dreaded "OOPS - ProDOS error code -> 40". •••

ZBASIC USERS RESPOND TO ZEDCOR'S CONTROVERSIAL INTEGER EXPRESSION EVALUATION METHOD

We have received a number of comments about ZBasic's unique method of interpreting equations. Space does not permit printing all of the hundreds of letters we received concerning this issue. Some of the letters follow:

Dear Zedcor,

After many frustrating hours, I became very impressed with the language and feel that it is vastly superior to other compiled BASIC languages that I have worked with. The one major problem that I have encountered is the calculation procedures in mixed mode arithmetic calculations. On several occasions, the lack of a number, or variable, at one of the extremities of an expression has forced a "real" variable to take on an integer value and has resulted in an "out-of-range" error. As you are aware, that error is not flagged and results in invalid output for unsuspecting users. In a complex program the error is extremely difficult to isolate.

Gary Saunders, DBA, CPA
North Canton, OH

The worst that could happen if I got floating point by accident would be some loss of speed. Compared to getting the wrong answer that's not too bad.

Dear Zedcor,

Why not let the programmer decide his choice of variable — you've built it into the language, it's just not used the way an intuitive user might guess.

Ed Capen
Dallas, TX

Dear Zedcor,

In your newsletter you asked for comments on forcing floating point math if any of the expressions in the equation are floating point.

I would surely like to see this incorporated in ZBasic.

I work mainly with small differences of large numbers. Sometimes I get caught by the forced integer even though I look for it in my programs. I can call for integer calculations if I want them. The worst that could happen if I got floating point by accident would be some loss of speed. Compared to getting the wrong answer that's not too bad.

I use ZBasic on both MSDOS and TRS-80's and look forward eagerly to the new MSDOS version.

Donald B. Fitz B.M.E., P.E., L.S.

Dear "Z" Newsletter,

The problem you discussed on page 18 of your winter 1986, 1987 newsletter, "ZBasic Equations Versus BASIC equations", is so severe I gave up on ZBasic for serious number crunching programs. I believe the default type for a numeric variable should be a floating point type.

Perhaps some statement could be used at the beginning of a program to change the default.

Harry W. Wiant, Jr.
Professor of Forestry
West Virginia University

Dear Zedcor,

As a a ZBasic programmer since October 1985, I was very happy to see the first edition of the newsletter.

You asked for comments... here are mine.

Not being a programmer nor having any aspirations to becoming a programmer, I purchased ZBasic because it appeared to be a way someone such as myself could develop commercial programs without having to take the time to learn C or some other language. I was working on a lengthy program in MBasic and was searching for a reasonably priced method of compiling it. ZBasic came along at just the right time and despite some difficulties, has enabled me to complete the project. Prior to starting this project in MBasic, my only programming experience was in one semester of FORTRAN while studying economics.

I would think your greatest potential market lies with people like me rather than with those who are more inclined toward becoming professional programmers. The more you can do to give us dummies access to programming power that was once only accessible to the elite programmers, the better.

For me, ZBasic has two enormous flaws that have caused me to lose tremendous amounts of time. The first is the way it handles perenthetical expressions in equations. My program has many complex equations and since I am not the world's greatest mathematician, I have lost days at a time trying to determine whether the wrong answers I was getting was due to my setting up of the equation or the way that ZBasic evaluated the equation. After much frustration, I eventually took to writing the equations in MBasic to test my math and converting them to ZBasic and tinkering until I got the right answer. In my opinion, it would be far better if ZBasic forced floating point whenever a floating point variable was encountered or whenever it was configured for floating point. Once I know that I am getting the correct results, I can then evaluate the equations t see whether I could speed things up by using integers. (In my applications, is almost always needed.)

The second major flaw is that I have on several occasions lost 30K and larger source code files by typing in the wrong extension at compile time. This could be avoided if ZBasic were to default to the .BAS extension for all source code files, .CHN or .COM for all chain and object files.

One other thing is the ability to handle larger programs and variables than 64K.

David Houston
Roselle, IL

(David, your suggestion for .BAS as an extension is now configurable in version 4.0. Thanks for the feedback).

Dear Zedcor,

You asked for feedback on your article "ZBasic Equations versus BASIC equations". I would like to cast a strong vote for your proposal to modify ZBasic so as to force an expression to be evaluated in floating point if it contains any floating point specifier.

Under the present convention a program may be logically and mathematically correct and yet lead to wrong answers. This necessitates extra time-consuming editing. Every algebraic expression in the program must be subjected to painstaking examination to determine how it will be interpreted by ZBasic.

Ideally when the user designates a default option of either integer or single or double precision floating point, it should apply to all expressions, not merely to the variables type. This would avoid the pitfalls for programs containing expressions that are properly, floating point while retaining the speed advantage for those programs in which integer default is appropriate throughout.

Robert Levit
San Francisco, CA

Dear Zedcor,

Feedback on the evaluation of ZBASIC EQUATIONS versus BASIC EQUATIONS, specifically the form (2.2+2.2)+(1.1+1.1), given in the newsletter:

Many ZBasic users are professionals who use more than one language for programming. Virtually all other languages (such as FORTRAN, COBOL, PL/1, PASCAL, C) are consistent in the evaluation of the above statement, and I believe it would be wise for ZBasic (if it is to be considered a serious language) to adhere to this "Standard" evaluation terminology.

Given A+B, A-B, A*B, or A/B:

If A and B are both integer, the result is integer; if either A or B is floating point, the result is floating point with the precision of the result equal to the highest precision of A or B.

Under this rule, the intermediate result (2.2+2.2) would be real (value 4.4). Note that this is not the same as forcing the entire statement to be floating point if any floating point number is encountered (this difference occurring with division). Thus:

(2.2+2.2)+1/2=4.4 (not 4.9)

This methodology give predictable results, is consistent with other languages, and should result in faster execution when converting the entire statement to floating point.

Dr. Walter S. Snyder, Ph.D
Professor of Engineering
University of South Florida
St. Petersburg, FL

"I recommend your product for its fantastic speed. To slow it down would be silly-- all someone has to do (if they don't want to pay attention to details) is buy a version of Microsoft BASIC. (and then suffer through the interminable waits associated with MSBASIC programs!)"

Dear Zedcor,

Please don't adapt the convention discussed on page 18 regarding the interpretation of statements with both integer and floating point expressions. Specifically, don't sacrifice speed when all that would suffice is a succinct explanation in the ZBasic manual as was printed in the "Z" Newsletter.

Although I have spent at least a couple of hours in the course of my program development projects trying to find out why certain expressions were truncated to integer form, your explanation has now cleared up my confusion. One of the strongest selling points of ZBasic to me and to all the people is execution time.

I recommend your product for its fantastic speed. To slow it down would be silly—all someone has to to do if they don't want to pay attention to details is buy a version of Microsoft BASIC (and then suffer through the interminable waits associated with MS-BASIC programs!)

Christopher A. Munger
Fairborn, Ohio

Dear Zedcor,

As I have noted to you in prior correspondence, I like speed. So I would not vote for the option you have noted on page 18 about forcing an entire equation to "real". But on the other hand, if your view of what your customer base needs is best supported by making such a change, I would not cast off ZBasic. On the other hand, I would like to see larger integer capability as Borland is now advertising for their new BASIC product. An approximately 32K limit is a great nuisance in trying to utilize integer math to optimize program speed.

R. Alden Rhoads
Grand Junction, CO

ZEDCOR RESPONDS...

Starting in all versions, 4.0 and better, there is a new configuration option to allow you to define your own type of expression evaluation (details in next column).

NEW ZBASIC EQUATION CONFIGURATION OPTION WOULD HAVE MADE KING SOLOMON HAPPY

The new ZBasic 4.0 has an important new configuration option:

```
Optimize Expressions for Integer Y/N?
```

Setting this option to "N" allows you to configure ZBasic to interpret expressions like most other languages. Leaving it at "Y" (the default) sets expressions to default to integer (as before).

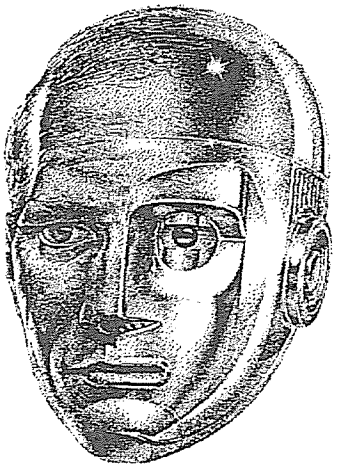
This option is in response to an article appearing in our premier issue, asking readers if they preferred ZBasic type equations, which offer improved speed, or other language type equations which offer less abiguity (Like FORTRAN and BASICA) .

Reader response was split, yet we felt there was enough response for both types that we have implemented it as a configuration option in all the new 4.0 versions.

For those unfamiliar with the article, it discussed the difference between the way ZBasic and other languages determined when to use floating point or integer math in an expression. (It is discussed in detail in the "Math" section of the new ZBasic manual for versions 4.0.) Readers were asked to respond with their preference.

The valuable feedback we received from you all has allowed us to make this important modification. We listen to your feedback.

Many thanks to those individuals that responded to the article, both to those that were printed here and to those that were not.•••



Dear Dr. Z

DEAR DR. Z,

I wish Andrew hadn't made ZBasic's file commands so simple and powerful, maybe we wouldn't have so many problems with them. I have been fighting them almost as long as he has been having fun writing them and haven't made that much headway yet. On the CP/M version I had to change all the WRITE# to PRINT# to get them to work.

This problem I am having is that if you enter one record then by using TYPE command from DOS you will see some of the item duplicated. If you put in several records then maybe the last record will show an extra line or maybe scattered parts of lines.

Either way it spells trouble because with two index files and using a file linking it can get licked and I sure won't be able to finish this program on schedule.

Since you can field some of these problems in your sleep maybe you can explain some of them to me.

George Snell
Port St. Lucie, FL

"I wish Andrew hadn't made ZBasic's file commands so simple and powerful, maybe we wouldn't have so many problems with them."

Dear George,

ZZzzzzzzzzzzzzz... Oh...that problem again! The problem you describe is common on a number of disk operating systems. Some systems, like CP/M, always save files with even record points (128 byte records with CP/M). ZBasic will also do something similar when doing OPEN"R", since it does not know where the end of the record is (you can position data anywhere within the record using RECORD.

When using Sequential file types OPEN"O", a 1A hex is saved as the last character and signifies to ZBasic (and to CP/M) that the end-of-file has been reached.

One way to try and visualize file types is to think of Random files as a number of contiguous records of exactly the same size. Say that each record is 200 bytes long and you save 180 bytes of data to that record. The last 20 bytes of the record will contain whatever extraneous data that was last in the disk buffer. If you don't pad the last 20 bytes with spaces or zeroes or whatever, they will contain whatever was left in the buffer.

It is a good idea to think about the way you want data to appear in the disk buffer. If you have extra space in the record you are probably not being as efficient as you could with disk space. In the example above, a file with 2000 records would have 40,000 bytes of wasted space.

Another problem you appear to be suffering from is confusion of when to use WRITE# and when to use PRINT#. In most cases the PRINT# statement should be used only for sequential type files. It is difficult to use with Random files because you cannot easily determine the number of bytes that are required for data. For example;

Let's look at the space required to save the numbers A#=12345.6789, B%=2 and C%=32000.

When using PRINT#1, A#","B%","C% PRINT# takes 12 bytes for A#, 3 bytes for B% and 7 bytes for C% plus one byte for each comma and one or two bytes for the Carriage return/linefeed (no linefeed on some operating systems), for a total of 25 or 26 bytes.

With WRITE#1,A#, B%, C% it is much simpler. Double precision always takes eight bytes* and integer always take two bytes for a total of 12 bytes required (*Normal configuration)

To make your decision easier, WRITE# is faster than PRINT# and READ# is 3-5 times faster than INPUT#.

It's really no wonder you're having such trouble. You're trying to fit a round peg in a square hole and it just won't cooperate.

Take a look at the "Files" section of the ZBasic reference manual for more in-depth examples.

My gut feeling is that you're really making this file business more complicated than it really is! If you just think of getting or putting bytes in or out of a file, which is all READ#, WRITE#, PRINT#, INPUT#, LINEINPUT#, do, and that the RECORD statement merely positions the file pointer for reading or writing bytes if you want (the file pointer moves along on it's own if you don't use RECORD), you may understand how easy file handling really is.

DEAR DR. Z,

You might be interested in the enclosed tests of ZBasic, comparing the different levels of precision and also comparing it with BASICA (version 1.1) on an IBM PC with an 8088 processor.

My gut feeling is that your really making this file business more complicated than it really is!

The precision of ZBasic is admirable, especially if configured to 54 digits. However I am absolutely surprised by the low speed, compared with interpreted BASICA. This contrasts with my experience with many production runs, where ZBasic was usually more than twice as fast as BASICA and sometimes nearly as fast as Turbo-Pascal with an 8087 coprocessor. However, none of the runs used trigonometric functions, only occasionally using LOG or EXP but extensive arithmetic. Does ZBasic have very precise, but slow routines for the transcendental functions?

The interesting thing is that with the two Microsoft versions of BASIC doubling the precision increases the computing time only a little, whereas for ZBasic it nearly doubles it.

Hans C. Joksch
West Hartfield, CT

Dear Hans,

The perplexity you are encountering is a common one. The ZBasic BCD (Binary Coded Decimal) floating point functions are designed

to return the correct answers (the trade-off is in speed). Microsoft uses Binary floating point and is designed more for speed (and the trade-off is in accuracy).

Example of Binary versus BCD accuracy:

	RESULT OF 1.2345 / 3.1415	RESULT OF 1.2345*3.1415
*	3.87818175	2.5447549615228837586
**	3.87818152958107	2.54475509997551

•ZBasic: Correct answers returned. 20 digits precision.
••MSBASIC. Wrong answers returned in double precision. Approximately 18 digits precision.

Accuracy test courtesy of
***Computer Language Magazine*:**

```
DEFSNG A-Z
WHILE X<1000000!
  Y=1!/(1-(X-1)/X)
  LPRINT "Result=";Y,
  LPRINT "Should be:";X,"Error=";X-Y
  X=X*10
WEND
```

MSBASIC ACCURACY RESULTS FOR BINARY MATH

Result=9	Should be: 9	Error=0
Result=89.99997	Should be: 90	Error=2.365112E-04
Result=900.0169	Should be: 900	Error=1.690674E-02
Result=8995.826	Should be: 9000	Error=4.173828
Result=89717.73	Should be: 90000	Error=282.2656
Result=883011.4	Should be: 900000	Error=16988.62

Note: ZBasic returned all correct answers for this test.

Nevertheless, your speed tests suffered from one major flaw. When comparing speeds for ZBasic floating point, you must configure the Double Precision accuracy to the desired accuracy NOT THE SINGLE PRECISION. This is a common misunderstanding. Speed increases of 5-7 times are realized when double precision is configured from 14 digits, down to 6 digits (which is what BASICA's binary floating point routines use for transcendentals).

Even so, transcendental functions go through a lot of work. So much work in fact, that the speed advantages of a compiler become less obvious since so much of the execution time is involved in complicated machine language computations. ZBasic even continues computing out the number of digits required to return the most accurate number.

While the speed variations are small in floating point when comparing compilers and interpreters, the accuracy variations are large. You need to decide if you want correct answers or speed.

Zedcor is preparing high-speed binary math options for future versions of ZBasic. These options will also utilize coprocessors, if available, and should be blindingly fast (but will suffer somewhat from the accuracy problems of other floating point packages).
Keep your eyes open for upgrade notices around August.

> ZBasic floating point is designed to return the correct answer...

Dear Dr. Z,

I have had a chance to use ZBasic for about three weeks now and have the following comments. First I think the package is a good buy, and I got a lot performance for the money, however I found a few aspects annoying having used Microsoft and BASICA for quite a while:

1. Is there any way to program entirely in lower case? Selecting the "Convert to uppercase" relieves the use of the Caps key, however it is very easy to create a BASIC token within it.

2. I really think graphic viewports and windows are needed.

3. I feel ZBasic should support the software function keys. KEY ON, KEY OFF etc.

4. I wish you didn't convert apostrophes to REM. I like apostrophes better.

5. I would like to see a CTRL-S toggle for starting/stopping program listings.

6. TRONB doesn't break out of an INPUT.

7. I want to see 8087 support.

Steve West
Tucson, AZ

Dear Steve,

Thank you for your comments. We're always looking for ways to improve our product and we can only do that when we get feedback from our customers.

Let me address your comments one at a time:

1. The problem can be solved by configuring BOTH "Convert to uppercase" AND "Spaces between Keywords" in version 4.0

2. Watch for upgrade notices late summer.

3. See MSDOS appendix for version 4.0

4. Fixed in version 4.0

5. ZBasic uses the Space Bar key to do this.

6. TRONB only checks for the break key at the beginning of each line. This is explained in the manual under TRON.

7. Late Summer watch your mailbox for upgrade notices (probably $19.95).

Dear Dr. Z,

Today I want to confirm that I am very satisfied with your ZBasic version 3.02 compiler, used on the IBM XT. It's speed surpasses Microsoft's BASICA considerably.

However I have some questions and wishes which cannot be answered by my Swiss distributors.

1. How can I create my own set of characters for the Hercules board?

2. How can ZBasic be adopted to the VEGA board from Video-7. The resolution is 752x410 pixels. However the mode 15 command reaches only up to 620x200. The same problem arises with the EGA board.

3. How can the Hercules board be switched to graphics in ZBasic.

4. How can I adapt the visible window? The program reaches to -8192x-8192 to +8192x+8192. This is Great! The visible window goes now from 0,0 to 1024,767. It would be an excellent solution to have the window in the size 1024x1024 as most monitors allow this dimension.

This is possible with the ZBasic "Macintosh" , why not with the ZBasic "IBM"?

5. How can the enormous text and printing possibilities of the Macintosh be transferred to the ZBasic "IBM"? Possibly by a 68020 supplementary board or by addition of BIOS eproms?

6. Unfortunately the German vowel-mutation cannot be processed, neither after the REM command. Besides that, the command UCASE$ does not work reliably with these characters. The ASCII up arrow symbol is missing.

7. I would like to have a more capable editor, similar to Turbo Pascal + Turbo BASIC. It should be possible to process complex strings.

8. Please send me a manual for developers. My Swiss distributors cannot supply it

Many thanks for your help,

Ludwig Thum
West Germany

Dear Ludwig,

In order...

1. ZBasic 3.02 does not have facilities to support Hercules graphics boards. Version 4.0 does. By following the instructions in the Hercules manuals for modifying your character sets you should be able to add the German vowel mutations easily.

2. Version 4.0 has facilities for doing all the EGA, CGA, Hercules and Monochrome modes. There are now modes 16-20 for the additional modes. Again, looks like you'll need version 4.0 to solve the problem.

You might also note that you can determine what graphic board is installed by using the new **CARDTYPE** statement.

3. Mode 20 in version 4.0 sets graphic output to the Hercules or Hercules plus boards or compatibles.

4. Version 4.0 of ZBasic for MSDOS and IBM computers allows you to set the relative coordinates using the COORDINATE or COORDINATE WINDOW statements.

This will allow you to set the relative coordinates to 1024x1024 if you choose. Note that most monitors have higher horizontal resolution than vertical. This is why we default to 1024x768.

5. As the new "Window" environments introduce new "Toolboxes" to enable various fonts and text styles as well as graphic commands, ZBasic will take advantage of them and make them available. Wouldn't it be nice to have compatibility for these things between computer lines?

6. The problem you may be seeing lies in the fact that mode 5 and 7 use their own character sets. Perhaps switching to mode 2,3,4,6 or some of the other BIOS text modes will solve your problem. We are looking at matching up the European characters in the internal set.

7. Version 4.0 comes with a full screen editor.

8. At this time there is not a special manual for developers. You may be referring to the "Developers Pack" we have advertised. This is just the regular ZBasic manual and six versions of ZBasic; MSDOS, Macintosh, Apple Dos 3.3, Apple ProDOS, CP/M and TRS-80 versions. All versions are bundled for a total of $195. Save $100.

Dear Dr. Z,

The first edition of the ZBasic newsletter included patches to MSDOS version 3.02 on page 14. Patching from &CF to &C7 will not work correctly when using a Tandy 2000. I will either have problems reading files or most likely hang the system.

In reference to John Brouwer's Quick Sort Routine, could you explain how to modify it to get a two level sort? For example, say each employee is given a department number between one and ten. We want the employee records to sort alphabetically within depart-

ments. The department #1 employees would appear first in alphabetical order, then the department# 2 employees, etc. Using Mr. Brouwer's Quick Sort, we sorted by department number and created a "key" file (record numbers in department order). Then, when we perform a second pass, how do we prevent a swap if department numbers are different?

John M. Reynolds, CPA, PC
Seymour, MO

Dear John,

The patches described in the first edition newsletters pertain to only certain versions of 3.02. As we discover the bugs they are fixed and ZBasic is reassembled. This changes the values at the described addresses. When you see patches in the newsletter, the number to change must be exactly the same as specified. If it is different, then your version has either been fixed already or is an older version. In either case DO NOT PATCH WHEN VALUES DIFFER!

As far as sorting by department, there are a number of options. One easy way would be to add a prefix to each name that specifies department number, in order, than to strip the department prefix off after sorting.

For example, say the string array you are sorting is Nme$(n) and the department number is stored in another array, Dept(n) (assuming less than 256 departments). Use this routine to force arrangement by department:

```
REM n= number of elements in the array
x=0
DO
  Nme$(x)=CHR$(Dept(x))+Nme$(x)
  x=x+1
UNTIL x=n
GOSUB"Quick Sort"
:
x=0
DO
  Dept(x)=ASC(Nme$(x))
  Nme$(x)=RIGHT$(Nme$(x),LEN(Nme$(x))-1)
  x=x+1
UNTIL x=n
END
```

In some cases, where only small sorts are required, the Shell-Metzner sort is probably easier to modify for multiple item sorts. In those cases you can just change the line with the SWAP statement on it to include all the elements of the sort.

Macintosh Questions

Dear Dr. Z,

I can't figure out how to add a "command Q" keyboard equivalent to my Macintosh menus. A Resource file similar to the one you have for icons might be helpful.

Also the new dialog, (up and down arrows), functions seem to work well with my 512K Macintosh and separate numeric pad. The arrows are much better than tab keys for entering numbers.

Doug McLellan
CentreVille, VA

Dear Doug,

To add control key equivalents to your menus simply precede the menu string contents with the "/". For example:

```
MENU 1,0,1,"File"
MENU 1,1,1,"/QQuit"
```

The second menu item would contain "Quit" and you could also invoke that item by pressing "command Q".

The new edition of the ZBasic manual is much easier to understand for the Macintosh. The entire appendix was rewritten and the the com-

mands have been alphabetized for easy reference.

Dear Dr. Z,

I would like to modify data in an applications data fork while the application is running. Is there a practical way of doing this in Macintosh ZBasic.

Also, I would like to write my own disk catalogue routine. Short of going into machine language, is there a way of listing HFS folder contents? I've been unable to access the folders with the version 3.03 DIR command.

Bill Chenault
Shalimar, FL

Dear Bill,

You may modify the data fork of applications by simply opening the application using the OPEN statement. The data fork is the default and you can modify it like any other data file.

Likewise, the resource fork of data files can be manipulated by appending an R to the OPEN types; RR for random, RI for input, RO for output, etc.

To get a directory of the files on either HFS or MFS systems, we have included a simple program on the master diskette called: NEWFILE.BAS. This example program shows you how to get the pathnames of all the files. Once these are loaded into a string array you can use them anyway you choose. Try running the program. Also see FILES$ in the new reference manual for a description of how to write your own routines to do this.

Hello Dr. Z,

I have a few problems with the ZBasic Macintosh version.

1. I can't eject a disk from the program line EJECT "ZBasic 3.01™", but in a local command, I can. Why?

2. How can I fix the disk so that every time I Quit from ZBasic, I can go to the finder?

3. I would like to vote for the establishment of a BBS but at many places. (I am in California). Long distance is too much money. GEnie is good. CompuServe is OK too.

4. Who is John Gillet? (in newsletter 1)

5. User Group president: Why not a technical officer from Zedcor to start with?

6. Does $19.95 apply to manual upgrade also?

7. The newsletter suggested to use System 3.2 and Finder 5.3; does this apply to Macintoshes 512E , 512 and plus?

8. When will you have further support in sound commands? It's kind of lonely to play music notes in monotone...

9. If I have a straight Mac and a SANE pkg. that is compatible with the 68881 coprocessor, do I need to get this processor in order to run the SANE package?

Anonymous

Dear Anonymous,

I will answer your questions in the order you gave them...

1. In version 3.02 and later the EJECT command was changed to a statement. The syntax is EJECT 1 or EJECT 2 where 1 is the internal and 2 is the external disk drive. This statement can be used either in direct mode or in your program. Note in the manual that **command** means you must do it from the standard line editor direct mode(use EJECT-n to "Unmount" the volume).

2. Create a "System Disk" from your system master. See your Macintosh user's guide for doing this. Then copy the ZBasic files from the master ZBasic diskette to your own system diskette. A Minifinder was included to allow us to include more files on the diskette. In version 4.0 there is not even a mini-finder since that was the only way we could fit the program on a 400K diskette.

3. We are currently on GEnie and you may leave questions. We will get back to you as soon as we can. Send your mail to: **ZBASIC**.
We are looking at CompuServe.

4. John Gillett is a friend of Apple. A long time technical manager and support person stationed out of Phoenix. He has been helping Apple dealers and is often seen at the Apple booths at trade shows all over the country. He also has a bulletin board in Scottsdale if you want to get a hold of him. (602) 951-4214. Read "TECHIE STUFF".

5. We are currently compiling the User Group list now. Our thoughts were that we should put a number of these important questions on a ballot and let the group decide.

6. No. Manuals upgrades are separate. For your information, if you don't know about it already, version 4.0 for the Macintosh is currently available, upgrades are $39.95 plus shipping. This includes a completely new manual. This manual does not apply to older versions since many new commands have been added and lots of things have been updated.

7. We recommend the newest Macintosh HFS systems whenever possible. There are cases when you cannot use the newest, such as with the older Macintosh 512K without the new ROM upgrades, the MAC XL, the LISA with Macworks, etc. In these cases use the latest MFS system. The version numbers vary. Contact your dealer for the latest version. With HFS avoid systems prior to 3.2. There were some major bugs that ZBasic is extremely sensitive to with systems like 3.0 and 3.1(you will become very familiar with system bombs if you use these versions of HFS).

8. Four voice sound is now fully supported for versions 3.03 and later. New statements like WAVE, SOUND WAIT, SOUND RESUME, SOUND function have all been added to give you the capabilities you (and others) requested.

9. No. With the upcoming ZBasic SANE option, the program will be intelligent enough to determine if the processor is there, in which case it will be used. If it is not available, the program will use the internal software routines.

It is important to do this otherwise programs sold commercially would not function on some machines, or two versions of each program, one with coprocessor support and one without, would have to distributed.

Dear Dr. Z,

I have appreciated the technical support that your company, in particular, David Lewis, has given our software development project.

When activating another window the active edit field in the active window does not de-activate. In addition the insertion points are lost when returning to an edit field. Please advise if the

edit field problem will be corrected in version 4.0

John H. Morton
Hawthorne Rubber Company

Dear John,

Yes (I've always wanted to write a "Dear John" letter).

Dear Dr. Z,

Comment report about ZBasic for the Macintosh version 3.02:

1. DIR does not function correctly, I can't list the directory of the second drive nor by DIR1 or DIR2....

2. In the menu EDIT, the choice of MDS EDIT causes a fatal system crash with either HFS or MFS EDIT.

3. Access to International Utilities Package of the Macintosh Toolbox does not function.

4. I have tried to use the toolbox call FN RELSTRING as it is described in the ZBasic addendum but the result is always false

5. Is it possible to CALL procedures compiled with C, Pascal or Assembler?

I hope you find solutions to these problems quickly because I need to use these features and my interest in a BASIC compiler lies in the possibilities of using the toolbox in the Macintosh Plus.

Jean-Claude Deroubaix
Institut de Sociologie
Bruxelles, Belgique

Dear Jean-Claude,

I will answer your questions in order...

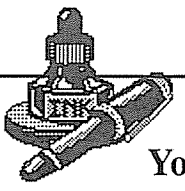
1. The DIR syntax is DIR "Pathname" for HFS systems and DIR "Root" for MFS. Pathnames are formatted with the Root directory as the first parameter and the path separated by colons: Root:folder:folder:folder
This was covered in the 3.02 addendum and is covered even more clearly in the new 4.0 manual and upgrade.

2. The MFS EDIT was removed from version 4.0 of ZBasic. The new built-in editor is incredible easy to use and is very powerful.

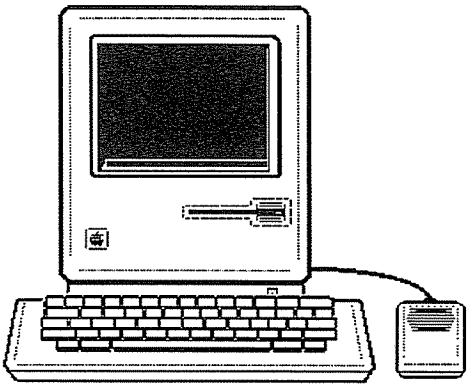
3. and 4. Access to ALL toolbox routines is now much easier and the documentation in the new manual should make using them much clearer for you.

5. You can execute other applications from ZBasic by using the RUN "filename" syntax. You may also wish to execute other applications and return to your ZBasic application with variables intact. To do this see the neat example in the 4.0 manual under WRITE FILE statement in the reference section.

We did a lot of work to make the new version of ZBasic work better with the toolbox and to make the documentation much easier to comprehend. The addendums to the addendums for the Macintosh version made it quite a chore finding information.



You can write Dr. Z, c/o Zedcor, 4500 E. Speedway, Suite 22, Tucson, AZ 85712-5305.



MACINTOSH

The new Macintosh version of ZBasic should be out by the time you read this. If you haven't ordered your upgrade yet call Zedcor today!!

New features include an all new manual, complete toolbox documentation, SELECT CASE and lots more.

ZBasic is now miles ahead of the competition (but don't worry, just cuz we're best, don't mean we're gonna rest!)

UNMOUNTING VOLUMES

One problem some of the Mac users have had is convincing the Finder to actual "Unmount" a diskette. Of course, you can use the regular EJECT 1 or EJECT 2 to eject a diskette, but the FINDER still asks you to re-insert the diskette now and then and that's a pain in the rear. To unmount the volume so the finder will not ask for it again use EJECT -1 or EJECT -2 in the new version 4.0.

NEW PARAMETERS FOR ADVANCED TOOLBOX PROGRAMMERS

After the manual went to the printer for version 4.0, Andrew had an idea for another little feature.

The manual states that expressions cannot be used as addresses in cases where VAR%, VAR&, VAR or VAR$ are used.

In order to use expressions as addresses you can now include a "#" in front of an expression (where the manual states you must use: VAR, VAR%, VAR&, VAR$). This symbol is not related to Double Precision when used in this context.

This should be a handy feature for some of you advanced toolbox hackers.

In the next newsletter we will describe some other, important, undocumented features (that were added after the manual was printed). You can now do SYSTEM and MEMORY MANAGER calls. (YEAH!!)

CALL FOR ZBASIC SUBROUTINES FOR THE MACINTOSH...

As we requested in the last issue we are still looking for programs and subroutines we can include in the newsletter. A number of readers did send in some nifty routines for doing a number of things. Please don't be shy! Any type of subroutine is welcome.

If you come up with some routines that solve problems please send them in. Your fellow programmers will be grateful.

We are currently compiling a number of subroutines, functions and programs for distribution. Jeff Moore is our "Master Librarian" so if you come up with some routines please make them to his attention.

SAVE-SAVE AS

One of the peculiarities of pre- 4.0 versions of ZBasic was the pain of having to answer prompts even when command-s was used. This has been fixed in version 4.0. Thank You Andrew.

PROBLEMS RUNNING EXAMPLE FILES

Some user's have experienced problems running the example files on the ZBasic master diskette. The problem is encountered when users change the configuration options. The programs on the diskette are configured for "DEFAULT" parameters. When you change things like "Spaces between keywords" or "Convert to uppercase", the programs will return syntax errors or function improperly.

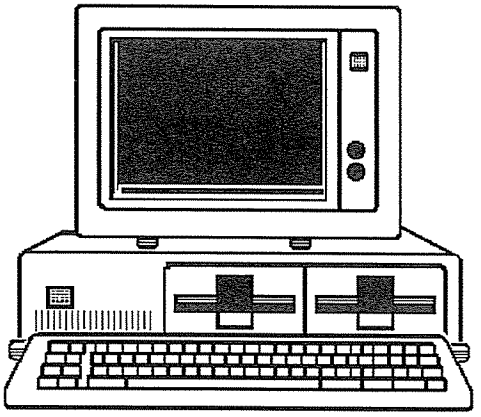
To fix this simply run the ZBasic from your original master disk which hasn't been configured to your preferences.

DESK ACCESSORIES

There has been a fairly large demand for a version of ZBasic that will create Desk Accessories. Keep an eye out for such a beast around the end of the year.

COMPILED SIZE

People want more compact code and a smaller runtime package. Ok. Look for something to make you happy towards the end of the year.•••



MSDOS IBM PC VERSION 4.0

The newest versions of ZBasic for PC's will knock your socks off. You should have it in your hands by the time you are reading this (if you don't already have your upgrade give Zedcor a call today!).

We are preparing some special tutorials for the next issue so keep a watch.

A BORED LAND OF MICRO SLOTHES

The competition is getting hot and heavy in IBM PC land. Many of you probably own a couple of versions of BASIC.

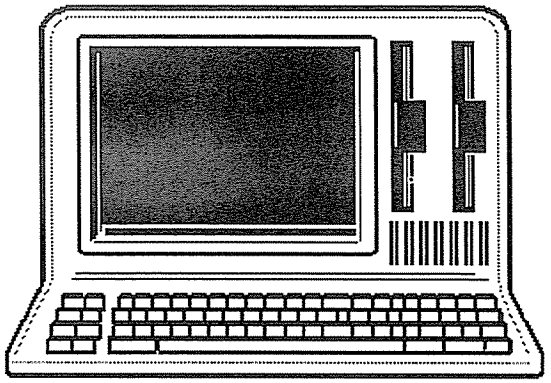
We're happy to report that ZBasic owners prefer the new ZBasic 4.0 over the competition ten to one. Most say it's not just the speed and ease of use that won them over, but Zedcor's attention to detail and consideration for programmer's needs that made the difference.

Thanks for your support folks!

CRITICAL ERROR ROUTINE TO TEST FOR "PRINTER READY"

Some of you have requested an example program or subroutine to test for "Printer Ready". The following routine was submitted by Chan Shippy. Thanks Chan! Chan also submitted some other routines that will be in future issues of "Z".

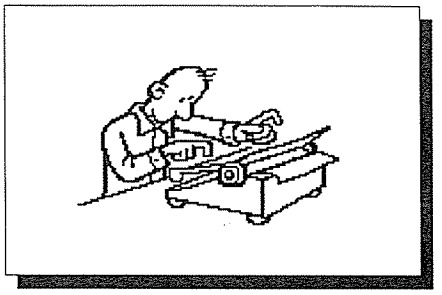
```
REM "PRNTEST.APP"
REM BY CHAN SHIPPY, RT. 1 Box 87
REM Colome, SD 57528
REM For IBM /MSDOS ZBasic only
:
CLS: MODE 3
GOSUB"Test Prn"
IF V$=CHR$(27) THEN END: REM Abort Printing
LPRINT "YeP! It Works!"
END
:
:
"Test Prn"   Printer Test Routine
LONG FN Testprn
  tst%=0
  MACHLG &50, &52, &BA, &00, &00, &B4, &02
  MACHLG &CD, &17, &D0, &D4, &BA, &00, &00
  MACHLG &81, &D2, &00, &00, &89, &16, tst%
  MACHLG &5A, &58
END FN =tst%
:
:
REM 1=Printer Ready,  0= Not Ready
LONG IF FN Testprn
  RETURN REM Returns if printer ON
XELSE
  SOUND 800,50: SOUND 600,50, SOUND 800,50
  LOCATE 0,24: CLS LINE
  LOCATE 10,24: COLOR 15,0
  PRINT"Printer NOT READY!"
  PRINT"<R> Retry,  <ESC> Abort Printing";
  COLOR 7,0
  "Try Again"
  V$=INKEY$: IF V$="" THEN "Try Again"
  LONG IF V$="R" OR V$="r"
   GOTO"Test Prn"
  XELSE
    IF V$=CHR$(27) THEN RETURN
    REM Check for <ESC> (CHR$(27)) on return
  END IF
SOUND 800, 130: GOTO "Try Again"
END IF
```



Z80 Versions TRS-80 and CP/M

The following program was submitted by Ira Goldklang. It reads a disk directory into an array for use within the program:

```
REM  TRS-80 1,3, 4 (in 3 mode) Directory
REM  Read Program
:
REM By Ira GoldKlang
:
REM This Program reads in a standard non-TRSDOS
REM Directory and places it into an array
REM FILE$(?) and sets NOFILES to the number of
REM files read in.
:
REM You must have DIMmed 15FILES$(70) in the
REM beginning of the program and set DRIVENO to
REM the disk drive to be read in.
:
REM Enjoy...
:
"READ DIRECTORY"
REM Input--> DRIVENO=Drive to read
REM Output-> NOFILES=# of files, FILES$(?)=DIR
CMD$="DIR";"+RIGHT$(UNS$(DRIVENO),1):CALL CMD$
START=15488:NOFILES=0
FOR COUNTER=1 TO 70:FILE$(COUNTER)="":NEXT
"LOOP":FOR ACROSS=0 TO 3
  FOR DROP= 0 TO 14
    CURR=START+ACROSS*15+DROP:HOLD=PEEK(CURR)
    LONG IF HOLD <> 32
      POKE CURR,191: HOLD$=CHR$(HOLD)
    END IF
  NEXT DROP
  IF LEN(HOLD$)=0 OR HOLD$="?" THEN RETURN
  NOFILES=NOFILES+
  FILE$(NOFILES)=HOLD$:HOLD$=""
NEXT ACROSS:START=START+64:GOTO"LOOP"
```



FREEWARE WORD PROCESSOR AVAILABLE FOR TRS-80'S

Gentleman,

Since ZBasic started out on a TRS-80, I thought your TRS-80 guru might get a kick out of my public domain word processor for the TRS-80 written in ZBasic. I assume you still have at least one TRS-80 back in the dusty recesses somewhere back behind all the IBMs!

The program is written entirely in ZBasic except for the character entry routine and screen scrolling, which are assembly language subroutines.

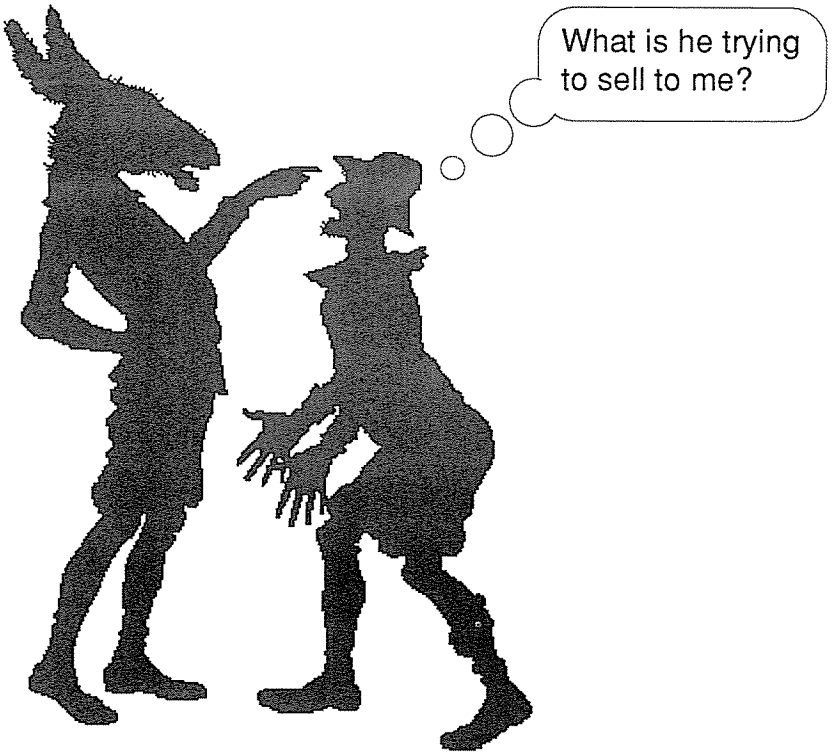
Use PWrite to print out the doc file. At the TRSDOS Ready prompt, type PW WRITE/DOC, then press Break for the command menu and select P to print. There is also an on-line help facility with a quick summary of the editing commands.

Anyone who sends me a blank disk and return postage is welcome to a copy. The current version can also be found on DL 2 of the TRS80PRO Forum of CompuServe.

Sincerely,

Pat Anderson
5420-324th Pl. S.E.
Fall City, WA 98024

Selling your Own Software make you feel like an -----?



LET ZEDCOR MARKET YOUR PROGRAMS!

We can do the marketing and you can do what you do best... *write programs that people need.*

No Program is too large or too small. The hard part is getting to an audience of people that will buy ***AND ZEDCOR KNOWS HOW TO DO THAT!.***

Zedcor pays top royalties for programs you create. Or, if you like, a one-time buy-out can sometimes be arranged.

Interested? Here's how to proceed:

• Send us a description of the program you've created. Let us know what it does better than the competition. Please DON'T send us a copy of your program (that comes later).

• If we're interested in your program, we'll send you a release form and more information about what you need to do next to become part of the Zedcor Team. Send program description to:

ZEDCOR, Software Marketing
4500 E. Speedway, #22
Tucson, AZ 85712-5305
(602) 881-8101

Letters to the Editor

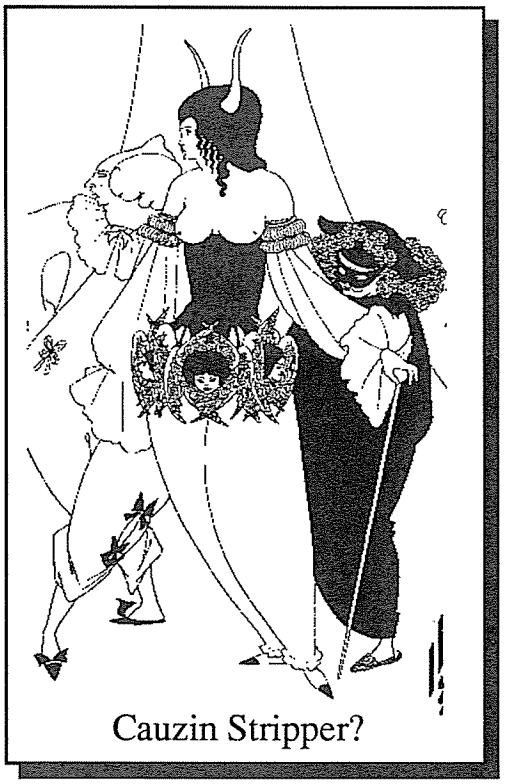
Dear Editor,

I received my free copy of the ZBasic newsletter today. I must say that I was most excited with the idea of the newsletter. I am also most interested in the idea of a ZBasic user's group. If such a group is started, I would like to join.

I want to also share with you and the others at Zedcor my pleasure in using ZBasic as the instructional computer language in my department. Being able to use the public domain versions of ZBasic (Apple &MS-DOS) has allowed us to move smoothly into a pluralistic computer environment. My department had dreaded the idea of this transition. ZBasic has made the transition not only smooth, but it has been exciting for both faculty and students. Many thanks for ZBasic. I must also tell you that my implementing ZBasic as a common language has made me a hero amoung the department faculty who teach computer related courses.

Something which you might want to share in your next newsletter is the idea of using the Cauzin Stripper to transfer ZBasic programs from one type of computer to another. So far we have had no problems using this technique as long as the programs did not contain any machine specific calls or machine language routines.

Now a technical question, what equipment/ program did you use to digitize the photos that appeared in the newsletter. One of my areas that my department is into is desktop publishing. Since the quality of the halftones is quite good, we are interested in the process used to achieve these graphic insertions.



Cauzin Stripper?

Please let me know when the ProDOS version of ZBasic is available. I would like to upgrade to that as soon as possible. Would Zedcor consider releasing a Public Domain ProDOS version? I would certainly like to use such a version in our department.

Al Rapp
P.O. Box 51
Blowing Rock, NC 28605-0051

Dear Al,

Thank you for your comments about using ZBasic in an educational environment.

We have been in contact with Cauzin and have several Strippers here. If there are other people interested in obtaining programs by this method please let us know. We could make them available.

We use a Macintosh Plus™, PageMaker™, a MicroTek™ 300 dot per inch scanner, Apple LaserWriter printer, and SuperPaint™ to produce the halftones for the newsletter. We have since switched to PageMaker™ 2.0. I also use the Radius full page monitor which makes things a whole lot easier to see.

The ProDOS version is available now! We also have a public domain limited version on GEnie for downloading (or send us a diskette and $5.00 for shipping and handling and we'll get you a copy).

Dear Editor,

Your first issue of the "Z" Newsletter was a welcome sight after the long wait. To do my part, I'm attaching some "Quick and Dirty" comments that have come to mind after reading the Winter 86-87 issue. I've also attached a copy of a simple function that I include in my programs when I want to be able to easily center text. Would you like more of such things? If all of us users would submit useful "additions" to ZBasic, we'd have even a more powerful language to work with.

Keep those upgrades and enhancements coming!

Other responses to issues discussed in the Winter issue:

• With regard to making a special "chopped down" version of ZBasic for desk accesories for the Macintosh mentioned on page 17, I am very interested! Please expedite development of such a product!

• I am another person who would like a much improved editor for all versions of ZBasic (I program in both the MSDOS and Macintosh versions).

The only item in IBM BASIC that I find very useful is the full screen editor which allows one to merely scroll back up the screen to previously listed statements and then modify them directly.

The one-line editor of ZBasic is much less efficient for the large-scale debugging required by quick hacks and large programming projects. In addition, the improvements mentioned in the pipeline for the Macintosh editor sound great.

The sooner you get a better editor developed the sooner you'll get a check for the upgrade!

• One very useful capability incorporated in the Microsoft BASIC compiler for the Macintosh (and one of the very few useful aspects of the new compiler) is the ability to embed a command into programs which is used only when printing program listings. This command sends page break commands to the printer.

This is very useful when developing modularized code in that subroutines, modules, and functions can be separated into stand-alone one- or two- sheet listings.

Program listing of CENTER TEXT function:

```
' FUNCTION:       Center_text
' AUTHOR:                Christophr A. Munger
' DATE:           February 4, 1987
' DESCRIP:
' This function uses these parameters to print
' the string center_text$ in the middle of a line
' whose width is specified by screen_width%.
' Note that line% is the line number to print on
' Note that screen_width% must be defined at
' the beginning of the program
'
' Variable:
'       Integer: line%
'                        line number to print on
'
'       String           center_text$
'                        Text to print
'
LONG FN Center_Text (line%, center_text$)
  textlen%=LEN(center_text$)
  PRINT @((screen_width%-textlen%)/2, center_text$);
END FN

Note: Change ' to REM for some versions of ZBasic.
```

Christopher A. Munger
Fairborn, Ohio

Dear Christopher,

Thanks for your comments and center FN:

• We're working on it!
• Version 4.0 for the Macintosh, MSDOS and Apple // ProDOS includes a new full screen editor. You asked for it you got it!
• We are looking at including your suggestion in future releases.

Dear Editor,

Enclosed is my subscription to "Z". Thanks for the first issue. I'll send a couple of articles in for it.

Comments on the first Newsletter:

1. Every article jumped from page to page. Reformat Z like MacTutor! Give each column a full page.

2. Include a known bug list with work arounds. This will help us avoid long hours of trying to get something to work when we should take a different approach.

3. Macintosh Ideas:

a. YES! We need desk accessories. No BASIC can do that (yet!). I bought TML Pascal 2.0 just to make desk accessories. What a pain to use Pascal when ZBasic could have done the job. Note that "Chopped Down" should only limit the code size to 32K.

b. You must optimize the compiler so that one line programs are not 32K. Look at the way TML Pascal generate code. It's tight.

c. New commands needed in version in 4.0:
   1. CASE statement
   2. RoundREC button
   3. Include a library statement
   4. Statement to call ASM from MDS/MPW
   5. A means to disable GetNextEvent calls the compiler makes.
   6. SLOT function for Mac II and SE
   7. GLOBAL and LOCAL statements to isolate variables. Once you do this then MAJOR projects can be done in ZBasic by many programmers without worrying about clashing variables.
   8. Provide a way to link ZBasic to dBase. Mac versions need to link to dMacIII (now sold by Nantucket). We are seeing dBase links to C and Pascal; why not ZBasic?
   9. DOWNTO, BEGIN, END, RECORD typing Pascal statements will help provide structure to ZBasic. (some changes will have to be made to maintain compatibility). Perhaps Pend, Precord, could be used to identify these Pascal statements from the current ZBasic statements.

D. Keep working on your editor but link QUED by Paragon CourseWare. It is HOT!!! I can configure QUED to transfer to ZBasic but I would like to see a way to transfer configure ZBasic to transfer to the editor of my choice

(EDIT was ok but QUED leaves it so far behind that once you use it you'll never go back.) without pulling up the SFdialog. Load and go is what we need. If you have never used QUED by a copy NOW!

While I'm taking about the editor, get rid of line numbers or provide auto line numbering or both.

Now that Microsoft has released their BASIC compiler (no, I didn't buy it even with their 1/2 price offer) Zedcor will have to match them feature for feature. When Borland releases their Turbo BASIC for the Mac the market will be crowded. Things like being able to make desk accessories, FKeys, including libraries, C Code etc., will determine who will survive,. Don't ever consider dropping your customer support or start charging for it. If you have to do something to offset the cost, raise your price to $99.95.

For the Future department:

4. Get forum up on GEnie!!! I'll help if needed.

5. Get a user's group going. Yes, I'll Help!

6. You might consider running your own BBS that would have source code available. An IBM clone with an 80 Megabyte drive should do it. 2400 baud is a must.

7. Rather than having ZBasic do all the toolbox calls, have a mode called EXPERT that would require the programmer to do all the manager calls, OS calls etc. This would put us in the driver's seat and avoid the sub-routine package. For example, what if I want to use the full screen for a game? Since you have initialized the Menu Manager it is hard to draw over the menu bar. Or maybe I want to invert the screen and port an IBM text program over. This option would make ZBasic the most powerful BASIC ever for the Mac. It would be worth another $75 to be able to do this. A programmer wants TOTAL CONTROL. TML Pascal has the I/O directive for "plain vanilla" applications.

8. How about a LOCAL GRAPHICS command (rather, a compiler directive) that would set everything up to the host machine graphic environment? On the Mac this would set everything to 512x342 pixels. I don't know of any machines that I can port my Mac Programs to with all the windows, dialogs etc. anyway.

I will be getting an IBM PC clone in a few months and I have recently acquired an Apple // + 64K machine. Any discount for buying two more ZBasic packages (perhaps without the manuals?)

Thanks again for a much needed product.

Anthony P. Oresteen
Batavia, IL

Dear Anthony,

Thanks for your comments. This kind of feedback really gives us ideas for future versions. I'll respond to in order...

1. Check-out this issue...

2. Future issues will include known bug lists and version number that the bug exists.

3A. Watch for special version later this year.
3B. Watch for an upgrade later this year.
3C1. Now in version 4.0
3C2. See example on page E49 of Mac appendix for putting default lines around buttons.
3C3. We are considering this feature now.
3C4. See CALL and RUN statements in the new manual. Also see example program under WRITE FILE.
3C5. Easy! Just don't turn on ANY events. Remember that things that enable event trapping are: DIALOG ON, BREAK ON, MENU ON, MOUSE ON (you can still USE ZBasic MOUSE

statements), TRON(any) and TIMER ON.
3C6. We are still waiting for Apple to provide the Mac II and SE seed machines they promised so we can begin implementing these features. Til then, we, and you, wait. Apple is more interested in seeding the larger software companies first. We have already implemented the SCSI ROM calls in version 4.0. See the new manual.
3C7. Look for future releases.
3C8. This is a good project for a third party developer to look into. Maybe you? If you do, we'd be glad to provide any support you need.
3C9. I am unsure why Pascal statements need to be added. We have the LONG prefix to indicate multiple line IF and FN structures. We have CASE, automatically indented DO/UNTIL, WHILE/WEND, FOR/NEXT/STEP and others. I'm not convinced most BASIC programmers want Pascal features just for structure. In fact, that's why some of us BASIC programmers Hate Pascal!
3D. See new editor in version 4.0. It's a real winner. Use TRANSFER command under the File menu to transfer to QUED. We've heard lots of good things about Paragon's QUED. We include one their brochures with ZBasic. Take a look at it.

Don't ever worry about us dropping customer support. We are in business to keep our customers happy.

Zedcor is currently on GEnie under "ZBASIC".

4. We're are currently on GEnie under ZBASIC. If we can get enough people to request a forum GEnie will set one up.

5. We'll include you on the ballot for President Anthony! Thanks for volunteering!

6. We are considering it. People interested keep us posted. We think that maybe a toll free line like GEnie might be more economical to most folks.

7. Version 4.0 allows you to do all of the toolbox calls. If there is a conflict between the compiler and the toolbox call it is mentioned in the newly revised toolbox chapter. There are lots of examples as well. And it's still as easy for beginners as it was before.

8. Most versions of ZBasic have the COORDINATE statement. This let's you set pixel or relative coordinates. It is one of the most powerful statements offered in ZBasic. This statement directly addresses your problem.

Thanks again for your comments.

Dear Editor,

I just received the first issue of the "Z" newsletter, it is very good! I enclose a subscription order and look forward to the next issue.

I am interested in ZBasic subroutines or utility library modules whether public domain or for sale.

We would also like to see a bulletin board from Zedcor, and am interested in any ZBasic bulletin Boards or user forums. (I need all the help I can get!)

Jerry Lawrence
Ft. Smith, AR

Dear Jerry,

Thanks for your comments. We are currently working hard putting together the subroutines we have here at Zedcor and the subroutines being sent in by readers of the newsletter. We will be making these available in the next issue of this newsletter and will add more as we go along. There are a number of routines included in this

issue too!

We are currently on GEnie under ZBASIC so feel free to send us mail if you want. We are considering adding our own bulletin board. The majority of readers seem to want us to provide public domain software and such on GEnie or some other toll free phone number. What do you think?

Dear Editor,

Enclosed is my subscription to "Z". I will hope that it will be a worthwhile item to receive. At the price of $19.95 it needs to be, as that is a more than many monthly magazines of demonstrated value. I might also point out that I am not in favor of having to pay to receive information as to the latest status, etc. of software that I own.

On the other hand, your software comes at a very competitive price, although the competition may get a lot tighter with the addition of Borland's new offering in BASIC (which I will obtain after the product has had a little time in user's hands to get initially customer debugged etc.) and "Z" may be a very attractive offering.

I am still awaiting the newest PC version of ZBasic. Hope it is coming along well (although a bit behind schedule as I understood it to be anticipated).

While I would like the product to be BASICA compatible, as noted, I am not sure just what you mean by that. I find it a nuisance that ZBasic cannot read or write BASICA binary files. Also that BASICA can either accept a blank or comma as a data delimiter whereas ZBasic insists on a comma. At least an alterable parameter to specifically select a delimeter as is available in General Electric Company timesharing would be desirable.

If your new product, including the screen editing, etc., is as capable for development as my combination of BDS and BASIC, then I would be happy to use ZBasic more exclusively. But, for instance, data files commercially purchased for BASIC access are typically space delimited or IBM BASIC random/binary, either way I have to reprocess the files to be able to read with ZBasic, and if then made random, have to have two separate files one each for each language. Fortunately, using a turbo accelerator board so speeds up my machine that I can afford the luxury of not bothering, in most cases, with random can just read ASCII files and drive on.

As I have said before, I think you have a great product. Better than current QuickBASIC or any other offering I have tried to date. But your IBM differences are a nuisance, particularly where the does not seem to be any great overriding necessity for such differences. i.e. in the latest upgrade why USR syntax instead of EOF (n). EOF does not seem to conflict with anything I can readily see already existing in ZBasic.

Anyway, keep up the good work, and will look forward to the newest version once released, and receiving the latest "Z".

R. Alden Rhoads
Grand Junction, CO

Dear Alden,

Thanks for your feedback.

As you read this you should already have version 4.0 in your hands. You will find that most of your suggestions have been implemented like Binary/BCD expression evaluation choice, EOF, more BASICA compatibility and others.

The problem you are having with file conversion is a real one and we are addressing it. We will make a public domain subroutine package available soon that will convert files for you.

Watch for future issues for this routine.

Dear Editor,

Enclosed is a check for $19.95 for one year's subscription to "Z". I was pleased to receive the first issue of "Z" even if it was "a long time a-comin" and rather chaotic; I hope that future issues will be better organized: nothing is, to me, more irritating than skipping from page to page, trying to follow an article which is presented in snippets. On a more constructive note, the "Customer program announcements" would, in my opinion, be more readable if the title or topic of programs were listed at the beginning of the line, followed by the address.

Attached are brief descriptions of four programs, written in ZBasic , which we currently offer. I would appreciate it if you would consider these for inclusion in the next newsletter.

Chris Whetton
Levittown, PA

"I was pleased to receive the first issue of "Z" even if it was 'a long time a-comin' and rather chaotic; I hope that future issues will be better organized; nothing is to me, more irritating than skipping from page to page, trying to follow an article which is presented in 'snippets'"

Dear Chris,

Thank you for your comments on our "snippets". You are absolutely right, of course, and I must admit I got carried away with PageMaker's ability to let text flow from one page to another. Rest assured this and future issues won't expose you to that type of chaotic reading.

We will include your program listings in the next issue.

Dear Editor,

Thank you for the copy of the "ZBasic Newsletter". I'm afraid it will be my last because I program very little and thus cannot justify the subscription price. However, I found much of interest in it. You want comments and subroutines. My offerings follow.

There are three subroutines you may be interested in:

1. Newton's method for determining roots
2. Displaying results in sig. digits.
3. Computing .5, 1 and 2% resistor values.

They are quite simple, and may have no value for you, so I'm not enclosing them. If you want them, just let me know. (*yes we want them for your library Bill! Ed.*)

I bought ZBasic because I was frustrated with the limitations of MBasic to six digits in all but plain arithmetic. ZBasic brought its own frustrations. Especially "ZBasic equations versus Basic equations", page 18 of the newsletter. Occasionally, a program of mine will go to someone else, or , as in the paragraph above, I will use someone elses program. Problems result. ZBasic is capable of accurate computations as well as speed. I think your decision for speed in that instance is the wrong decision. It sure slows down programming.

Another area I have a problem is limited string capability. As stated above, I program very little, so am not to proficient with strings. David Kuhn's unit conversion program is a case in point. He is proficient, and I think he may have

been showing off a bit. If you examine the program lines for displaying output in his original program and the way I changed it, you will see what I mean. I got all but the last output line converted, then fell flat on it because ZBasic had no comparable capability. The solution was to substitute one line for several and it works fine.

I would like to be advised if you ever change that parentheses thing. I will want an upgrade.

Dear Bill,

Be advised that 4.0 solves your problem with parentheses.

The problem you are having with strings is that ZBasic forces you to convert string expressions to use simple strings, instead of complex strings. While this may be nice for expert programmers in the interpreted mode, it makes string manipulations much slower. (Since many of you are aware that ZBasic string manipulations are 10 to 100 times faster than QuickBASIC and TurboBASIC, I'm sure most of you don't want it changed. Let us know if you do!).

> "I just received my promotional copy of "Z" today and am indeed interested. However, when I learned that it was only a quarterly publication, the $20 subscription seemed a bit expensive..."

Dear Editor,

I have just received my promotional copy of "Z" today and am indeed interested. However, when I learned that it was only a quarterly publication, the $20 subscription seemed a bit expensive, if the amount of information in the first issue is any indication of things to come. I have learned more about ZBasic (and more) reading MacTutor. So for your marketing information, I am not subscribing given your current price. (I do like the compiler though!!!).

Curt Black

Dear Curt,

Sorry you feel the price is too high. If we took advertising and had thousands of subscribers, we could offer it for much less.

MacTUTOR is certainly a good publication and we appreciate their support of ZBasic.

I think this and future issues will convince unbelievers of the value of this publication.

I aim to make it worth much more than the $19.95 we ask for it. Heck, even a couple of simple subroutines provided by other user's would save you enough time to make it well worth the money.

We're really not getting rich over this newsletter. $19.95 barely covers our costs for time, publishing and overhead.

Dear Editor,

If ZBasic is to be taken as a serious and sophisticated language, then many of the restrictions and or lack of appropriate commands must be removed. The following is a partial list of those items that readily comes to mind.

Variable names: The simplistic requirement that a variable name not have a keyword embedded within it is not only frustrating but also shows a lack of sophistication. It is true that one can always use lowercase but it is not always convenient or desirable.

ON INKEY$(x): ESCape should be added to the

list of keys supported by the ON INKEY$(x) statement.It should also be possible to use both the ON INKEY$(x) GOTO and GOSUB statement. It would also be useful to allow the user to add keys to the list.

Screen Line (CSRLIN): Because of frequent need to determine the current screen line of the cursor, a function for this purpose should be included within ZBasic. To have to use machine language subroutine is ludicrous.

FOR...NEXT loops: In keeping with the trend in all major languages, it is desirable for FOR...NEXT loops to be executed conditionally, i.e. the loop is to be executed only if the initial value is less than or equal to the terminal value (in case of positive STEP, visa-versa if STEP is negative). This has become a standard methodology in other languages because it has been found that in the majority of cases this is the intent of the programmer. If it is strongly felt that the old procedure should be kept, a new statement (e.g. FOR1) could be used.

LONG IF...XELSE...ENDIF: This extremely useful statement could be considerably enhanced by incorporating an XELSE IF statement similar to FORTRAN-77's ELSE IF. This structure can currently only be accomplished by nesting LONG IF statements which is aggravating at best.

ON ERROR GOSUB: This syntax is quite restrictive. ON ERROR GOTO should also be an option.

End-Of-File: It is inconceivable that an EOF function is not included within the ZBasic structure. Since this is one of the of the more common tests that must be made on file input, every major language has some form of this incorporated into it. The necessity for using ON ERROR GOSUB to determine the end-of-file is messy and a pain to the programmer. The use of USR1 (number) is also not desirable as it has no mnemonic attribute.

Directories: The OPEN statement should allow for access to files contained within directories other than the current directory. This is supported by MSDOS 2.x and higher. It is a significant restriction on ZBasic's flexibility for this ability to be omitted (perhaps version 4.0 corrects this). Along with this, CHDIR should be a function useable with the program as well as from the editor (even though this can be accomplished by using the CALL statement).

Append: Extremely useful would be an APPEND mode in the OPEN statement whereby data can be added sequentially to an existing file.

Size Restrictions: Major programs can easily exceed the 64K limit on program size; data frequently exceeds the 64K limit. There must be some provision for extending the size of programs and data segments. I also note that the maximum size of the INDEX$ segment is 65535 (not the top of memory as indicated in the manual and in the newsletter). Hopefully, this will be corrected in version 4.0.

Suggestions for consideration: these concepts may or may not be easily incorporated, but would make applications programming much easier:

Command line retrieval: Instead of having to read the MSDOS command line using PEEKs to find parameters that may be passed to the .COM program, why not incorporate a READ#0 (or some such) statement to return this information. The passing of parameters to a compiled program at the time they are invoked is a common desire among serious programmers.

Subroutines with arguments: One of the inherent weaknesses in the BASIC language is the inability to pass arguments to subroutines. This is especially notable in array manipulations and is one of the reasons that BASIC has not been

seriously used in engineering applications. The best way to pass arguments to a subroutine appears to be by address and not by value (especially when dealing with arrays). If such a subroutine using arguments could be incorporated into ZBasic, it would enhance its standing in the scientific community immensely. If the subroutine was structured to be an independent program unit such that variable X in the subroutine was different than variable X in the main program (i.e. non-global variables) the concept of building a library of routines would be enhanced.

Overall I believe that ZBasic is a good piece of software. With the proper enhancements, it can become a great piece of software.

Walter S. Snyder, Ph.D.
Professor of Engineering
University of Florida

Dear Dr. Snyder,

Thank you for your comments and suggestions about ZBasic for MSDOS machines. I will respond to the comments in topic order:

Variable names: This has been implemented in version 4.0. Simply set "Space between keywords" to "YES". This allows token parsing to function properly and allows you to insert keywords in any case into your variables.

ON INKEY$(x): We are looking at ways of implementing other keys in this situation.

Screen Line (CSRLIN): Included in version 4.0

FOR...NEXT loops: We will keep this in mind for future releases. In the meantime, look at WHILE/WEND. This loop structure checks for values at the beginning of the loop.

LONG IF...XELSE...ENDIF: ZBasic does allow XELSE : IF without the complicated LONG structure.

ON ERROR GOSUB: We will look in to your suggestions.

End-Of-File: EOF Implemented in version 4.0.

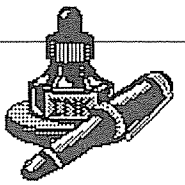
Directories: Implemented in version 4.0. Also MKDIR and RMDIR have been added.

Append: May be in version 4.0. Not finished at time of this letter (I told the MSDOS guys about it today).

Size Restrictions: Version 4.0 allows string and real arrays to 640K on 8088/8086 and one megabyte on 80286 machines. Source code and object code must still be chained when larger than 64K (watch for an upgrade later this year for unlimited source and object code size).

Command line retrieval: See COMMAND$ statement in version 4.0. This will return the command line for you.

Subroutines with arguments: We are working on this now. Watch for upgrades later this year.



Letters to the Editor may be submitted to: Michael A. Gariepy, Editor, 4500 E. Speedway, Suite 22, Tucson, AZ 85712-5305

Dear Editor,

Congratulations on publication of your newsletter. I am glad to see more support for ZBasic and am including a check for my subscription. You requested comments on your newsletter and indeed I have several. First DO NOT WRITE ANOTHER NEWSLETTER WITH THE ARTICLES SCATTERED ALL OVER THE PAGES. An otherwise great newsletter was seriously marred by this feature. Keep the articles on one or two pages to increase readability. If you publish like this it would be only fair to include a disk with an index so we could use our computers to use our computers to tell us where the articles are located (sarcasm intended!!). Second, it would be a good idea to include a disk with the newsletter with the source code for all the software published in the newsletter. This would allow you to publish larger programs without taking up space in the newsletter as it would only be necessary to provide documentation or even a simple description for people to use the programs. All programs for all machines should be included in this disk. As for why; First, people do buy new computers and it would be nice to have software for your new machine rather than having to order it, Second; even though the software might not work on a specific machine someone may modify or rewrite to work on their machine, and hopefull submit it to your newsletter. Lastly, friends also buy computer languages and several disk of free software is a very good reason to buy one language over another.

I like the Dr. Z column and you may count this as one vote to keep him. Also, while I am very interested in the users group my job requires quite a bit of time so I could not be of much use running the group. I would be happy to contribute either money and or programs for any users group.

Don't forget the CP/M user. While I plan to get an MSDOS portable and perhaps in a few years a UNIX or other multiuser system, there are no plans to retire my "old reliable" CP/M machines. In fact, I want software that will allow me to go from one machine to another with minimal trouble. That is why I use ZBasic instead of the other advanced Basic's now on the market. It would also be a good idea to include (at least once a year) in your newsletter a complete list of all operating systems and disk formats that ZBasic is available for. How about 3.5" MSDOS format and UNIX/ATARI 520/AMIGA availability?

I have included a disk in this package that con-

May "Z" force be with you!

tains the following files for inclusion in either your newsletter or public domain library:

ZTERM.SRC
A simple terminal program for the Osborne I and IV (Vixen) that overcomes difficulties with COM functions on these machines. It does so by resetting the I/O byte. This program also will run in MBasic without modification and can really show off ZBasic's speed. On my machine the program will run at 300 baud under MBasic and up to at least 9600 baud in ZBasic. It may go faster but I have nothing to check it with.

May "Z" force be with you.

David Postler
Egin, IL

Dear David,

You weren't the only one to criticize my layout. Oh well...

Your ideas about including diskettes are interesting but I cannot imagine how we could provide disk formats for everyone.

My idea was to provide listings of smaller programs in the newsletter and offer those, and other larger programs, on diskettes at a minimal cost. Say, $5-$10 a diskette. We would "clean-up" the code and test it to make sure it works. We would also be "Keepers of the Code" so-to-speak. A list of the programs available would appear in each newsletter and you could order them as you needed through our toll free order line.

We have no intentions of forgetting the old CP/M machines. They really are old reliables.

I am especially relieved to hear that Zedcor does not employ KGB and SS Software Confiscators. Nothing in the world like waking up in the middle of the night to find three guys in trench-coats ransacking your diskette boxes by flash-light. I can now sleep without fear of being caught between the ZBasic diskettes and a hoard of ruthless, take-no-prisoners, Zedcor-employed foreign agents.

Levity aside, I appreciate your responses and I am looking forward to using ZBasic.

(Name witheld)

Thanks to feedback from this person and others, our new licensing agreement has been revised to be less ambiguous. It really just says "Don't give away or sell ZBasic. We own the copy-right. You can sell software you create with ZBasic without paying us royalties or a runtime fee.

RESPONSE OF THE MONTH

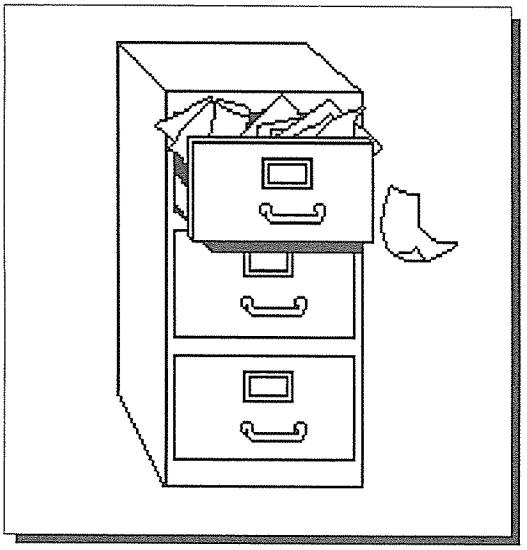
A gentleman and I had some correspondence concerning the ZBasic licensing agreement. He had some concern about the wording and wrote me a letter. I answered with my best non-legaleze. His response was very interesting and I include it here for your enjoyment.

Dear Mr. Gariepy,

Thank you for answering my questions about your licensing agreement for ZBasic. Now that I have a better understanding of the agreement I feel much more comfortable about signing it (a signed copy is enclosed).

I am especially relieved to hear that Zedcor does not employ KGB and SS Software Confiscators. Nothing in the world like waking up in the middle of the night to find three guys in trenchcoats ransacking your diskette boxes by flashlight.

LETTER OF THE MONTH



Dear Sirs,

Enclosed is my original ZBasic disk for upgrade to version 3.03. I would like to make a few comments and suggestions as fas as the newsletter is concerned. I thought the newsletter was well put together. Thanks guys.

The idea of a users group is also a great idea. It could be called ZUG (of course) and of course bugs in release 3.03 could be called ZUG BUGS. Now think of the possibilities...you could create a users group with the newsletter as the means of communication and subscription automatically qualifies the person for membership. Of course if someone purchased ZBasic and did not subscribe you could send a ZUG THUG to collect the 20 dollars.

And you could have an annual ZUGATHON; it would be a contest for the best program created with ZBasic. And you could give an award for the ZUGLIEST program as well.

This summer when I get a hard drive I will probably start a bulletin board for ZBasic people and I could call it ZUGNET, or maybe TELEZUG.

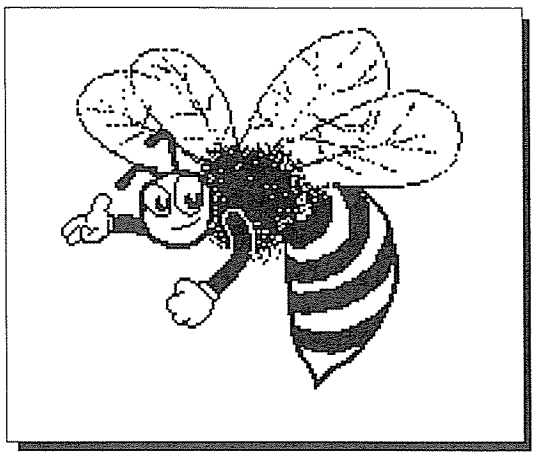
I might also suggest a collection of public domain ZBasic routines to put out on a quarterly basis as well as on diskette. You might charge 5 dollars per diskette also on a subscription basis; you could establish a list of the programs in the newsletter for those people that wanted to order disks individually, or your could make them available on TELEZUG. You might offer a free subscription to those authors submitting routines that are accepted in the library; this would encourage submissions (hopefully). You could call the disks ZUGETTES (of course).

And finally if you're tired of this letter you could quitely think..."I Wish this idiot would ZUGOFF!".

...a users group is also a great idea. It could be called ZUG (of course)... Of course if someone purchased ZBasic and did not subscribe you could send a ZUG THUG to collect the 20 dollars.

Thanks again, and I look forward to my upgrade. I am very satisfied with the program. However, now that I think about it, I do have a question that I seem unable to solve even with "Inside Macintosh". Every time a compiled ZBasic program runs the menu bar immediately comes up. I would love a command that writes and erases the menu bar and it would be nice if the menu bar defaulted to off.

My Personal Regards,
Sidney Beckman
Box 15162
Nacodoches, TX 75962



Helpful Hints

by
Jeff Moore
Zedcor Technical Support

Being the Technical Support person at Zedcor I receive a variety of calls about all kinds of subject matter.

One thing I have noticed is a lot of you have problems with files structure. I would like to suggest a couple of things to help you solve your problems. When I am working with files I always keep the following program handy:

```
INPUT "Enter File Name"; Filename$
OPEN "I",1,Filename$
ON ERROR GOSUB 65535
READ#1,A$;1 : REM PRIMING READ
WHILE ERROR = 0
LONG IF ASC(A$)>31 AND ASC(A$)< 128
  PRINT A$;
XELSE
  PRINT ".";
END IF
READ#1,A$;1
WEND
ERROR=0
CLOSE : END
```

This program allows me to look at the structure of my files and see if the same thing I thought was in there is in there.

Notice line 4 which is a Priming Read, this sets the WHILE construction to end in the correct place and not read a duplicate byte at end and doesn't PRINT an extra ASCII value on the screen.

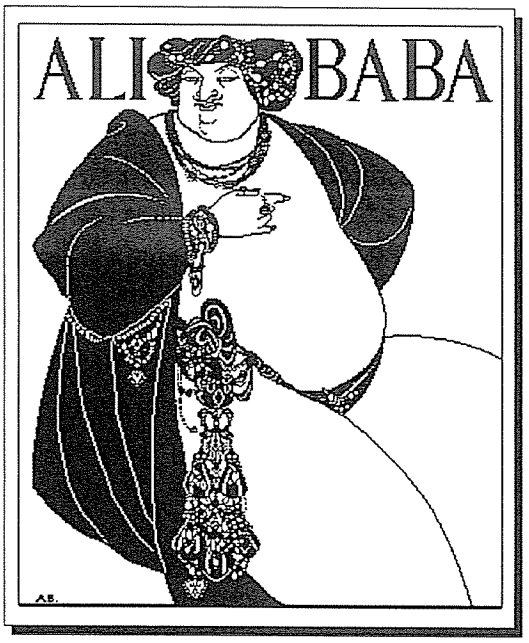
This could also be used with the DO UNTIL construct and Random or Sequential files and can be very helpful in figuring out what is going on in your file.

You can read any ZBasic file in this fashion including tokenized files (actually ANY file for that matter). This will allow you flexiblity in doing many things. You can see how the files are actually stored and write many utility programs to, say, strip out control characters or something.

One note on the program above: notice I convert control characters to a period so that they are not printed to the screen. This prevents them from screwing up the screen output.

You might want to add the ROUTE 128 statement to send output to the printer for a hard copy listing.

Sorry for the short column this month. Mike told me I had to write a column a day before deadline. Next issue my column will be much more comprehensive and informative.



Support Notes

As most of you know, David Lewis has left Zedcor. My name is Jeff Moore and I will be handling Technical Support. My background is a Degree in Computer Science and I have worked with college students for the past two years helping them with BASIC programming problems.

I would like to thank all those who called over the past three months and been patient with me during my learning curve. I've enjoyed your calls and letters.

To facilitate technical support calls I would request you have the following information when you call; license agreement number, computer type, version number of ZBasic, any special equipment or memory resident programs you maybe using and, last but not least, the way you have your version of ZBasic configured.

If you have a problem and we can't solve it over the phone, I will request that you send a disk in with the above information and the problems you are having with the program. I will look at it as soon as possible and send it back to you.

NOTES ON ProDOS

Now for some notes on the new Prodos 64k & 128k versions. On page D5 of the manual it states that the 128k version requires an extended 80 Column Card and a 65C02 or 65802 microprocessor. If you load your 128k version and you get a Prodos 28 error code, you have the old chip.

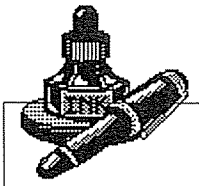
What you need to do is go to the local electronics store or Apple Computer dealer and pick up the new version of the microprocessor (they run about $13).

Another small problem with IIGS owners arose when using the file named Prodos on your system disk. That file is not really Prodos but a boot progam for Prodos which is named P8 on the disk. What you need to do for that is copy that file over and rename it Prodos and everything will work fine.

NOTES TO OTHER USERS

My apologies for the brevity of this column. My next column will be much broader and cover more of the problems for each specific machine.

Since many of the problems are repeated daily, I will discuss the more common ones.



Jeff Moore can be reached by pone at (602) 795-3996 from Noon til 5PM, Monday through Friday (MST).

He may also be mail at ZBasic Technical Support, Zedcor, 4500 E. Speedway, Suite 22, Tucson, AZ 85712-5305

COMING NEXT ISSUE...

• We'll have some entertaining source code for your amusement and teach you some ZBasic tricks.

• Super Powerful routines to determine day of the week, month of the year and elapsed days for any date. These routines are small and descriptions make it easy to use them in your programs.

• Macintosh: We'll have a special section on the new toolbox routines that were not covered in the new manual. Now you can do System and Memory manager calls!

• The continuing saga of "SUPPORT BLUES". New expanded format is something to behold.

• IBM, ProDOS and Z80 programs that you've been waiting for!

DON'T MISS IT!

WE WANT YOU TO SUBSCRIBE TODAY!

The "Z" newsletter provides you with important information about your language of choice. Just one or two of the programs and ideas provided can save you many hours of work and frustration.

Only $19.95 per year (save $5 off the newstand price) and only $37 for two years (save $10). Send in the coupon or call our toll free order line to subscribe with your credit card.

1-800-482-4567

Name_____

Address_____

City_____ State_____ ZIP_____

Daytime Phone_____

Credit card number _____ Exp.____

Signature_____

$19.95 One year___ $37.00 Two years____ (quarterly publication)
(please add $5.00 for Canadian and Overseas subscriptions)

Mail Order to:

"Z" Newsletter
4500 E. SPEEDWAY, SUITE 22
TUCSON, AZ 85712-5305

"Z" Newsletter
ZEDCOR, INC.
4500 E. SPEEDWAY, SUITE 22
TUCSON, AZ 85712-5305

Bulk Rate
U.S. Postage
PAID
Tucson, AZ
Permit Number
2220

SEND TO:

324